Package 'vprr'

July 12, 2022

Type Package

Title Processing and Visualization of Video Plankton Recorder Data

Version 0.2.3

Maintainer Emily O'Grady <vprrcontact@gmail.com>

Description An oceanographic data processing package for analyzing and visualizing Video Plankton Recorder data. This package was developed at 'Bedford Institute of Oceanography'. Functions are designed to process automated image classification output and create organized and easily portable data products.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Depends R (>= 2.10)

Imports ggplot2, oce, dplyr, interp, magick, gsw, tidyr, stringr, metR, gridExtra, lubridate, cmocean, withr, usethis

Suggests knitr, rmarkdown, testthat, markdown

VignetteBuilder knitr

BuildVignettes true

URL https://eogrady21.github.io/vprr/

NeedsCompilation no

Author Emily O'Grady [aut, cre, cph], Kevin Sorochan dev [aut], Catherine Johnson [aut]

Repository CRAN

Date/Publication 2022-07-12 14:20:02 UTC

R topics documented:

bin_calculate . . . 3
bin_cast . . . 4
concentration_category . . . 4
ctd_cast . . . 5
ctd_dat_combine . . . 6
ctd_df_cols . . . 7
ctd_roi_merge . . . 8
ctd_roi_oce . . . 9
getRoiMeasurements . . . 9
insertRow . . . 10
isopycnal_calculate . . . 10
normalize_matrix . . . 11
px_to_mm . . . 11
roimeas_dat_combine . . . 12
roi_dat_combine . . . 12
size_df_f . . . 13
taxa_conc_n . . . 14
vpr_autoid_check . . . 15
vpr_autoid_copy . . . 15
vpr_autoid_create . . . 16
vpr_autoid_read . . . 17
vpr_category . . . 19
vpr_category_create . . . 20
vpr_ctdroisize_merge . . . 20
vpr_ctdroi_merge . . . 21
vpr_ctd_files . . . 22
vpr_ctd_read . . . 22
vpr_ctd_ymd . . . 23
vpr_day . . . 24
vpr_dayhour . . . 25
vpr_hour . . . 25
vpr_img_category . . . 26
vpr_img_check . . . 27
vpr_img_copy . . . 27
vpr_img_depth . . . 28
vpr_img_reclassified . . . 28
vpr_manual_classification . . . 29
vpr_oce_create . . . 30
vpr_plot_contour . . . 30
vpr_plot_histsize . . . 31
vpr_plot_profile . . . 32
vpr_plot_sizefreq . . . 33
vpr_plot_TS . . . 33
vpr_plot_TScat . . . 34
vpr_pred_read . . . 34
vpr_roi . . . 35

vpr_roi_concentration 35
vpr_save 36
vpr_size_bin 37
vpr_summary 38
vpr_trrois_size 39
vp_plot_matrix 39
vp_plot_unkn 40

Index **41**

`bin_calculate` *Get bin averages for VPR and CTD data*

Description

Bins CTD data for an individual cast to avoid depth averaging across tow-yo's

Usage

```
bin_calculate(data, binSize = 1, imageVolume, rev = FALSE)
```

Arguments

`data`	ctd data frame object including scan, salinity, temperature, depth, conductivity, time, fluor_ref, turbidity_ref, turbidity_mv, altitude, cast_id, n_roi
`binSize`	the height of bins over which to average, default is 1 metre
`imageVolume`	the volume of VPR images used for calculating concentrations (mm^3)
`rev`	logical value, if TRUE, binning will begin at bottom of each cast, this controls data loss due to uneven binning over depth. If bins begin at bottom, small amounts of data may be lost at the surface of each cast, if binning begins at surface (rev = FALSE), small amounts of data may be lost at bottom of each cast

Details

Image volume calculations can change based on optical setting of VPR as well as autodeck setting used to process images For IML2018051 (S2) image volume was calculated as 108155 mm^3 by seascan (6.6 cubic inches) For COR2019002 S2 image volume was calculated as 83663 mm^3 and S3 image volume was calculated as 366082 mm^3. Used internally (`bin_cast`) after `ctd_cast` on a single ascending or descending section of VPR cast

Note

binSize should be carefully considered for best results

Depth is used for calculations! Please ensure depth is included in data frame using swDepth

Author(s)

E. Chisholm, K. Sorochan

bin_cast *Bin vpr data*

Description

Formats oce style VPR data into depth averaged bins using ctd_cast and bin_calculate This function is used inside concentration_category

Usage

```
bin_cast(ctd_roi_oce, imageVolume, binSize, rev = FALSE)
```

Arguments

ctd_roi_oce	oce ctd format VPR data from vpr_oce_create
imageVolume	the volume of VPR images used for calculating concentrations (mm^3)
binSize	passed to bin_calculate, determines size of depth bins over which data is averaged
rev	logical value,passed to bin_calculate if TRUE, binning will begin at bottom of each cast, this controls data loss due to uneven binning over depth. If bins begin at bottom, small amounts of data may be lost at the surface of each cast, if binning begins at surface (rev = FALSE), small amounts of data may be lost at bottom of each cast

Details

Image volume calculations can change based on optical setting of VPR as well as autodeck setting used to process images For IML2018051 (S2) image volume was calculated as 108155 mm^3 by seascan (6.6 cubic inches) For COR2019002 S2 image volume was calculated as 83663 mm^3 and S3 image volume was calculated as 366082 mm^3

Value

A dataframe of depth averaged bins of VPR data over an entire cast with calculated concentration values

concentration_category *Binned concentrations*

Description

This function produces depth binned concentrations for a specified taxa. Similar to bin_cast but calculates concentrations for only one taxa. Used inside vpr_roi_concentration

Usage

```
concentration_category(data, taxa, binSize, imageVolume, rev = FALSE)
```

Arguments

data	dataframe produced by processing internal to vpr_roi_concentration
taxa	name of taxa isolated
binSize	passed to bin_calculate, determines size of depth bins over which data is averaged
imageVolume	the volume of VPR images used for calculating concentrations (mm^3)
rev	Logical value defining direction of binning, FALSE - bins will be calculated from surface to bottom, TRUE- bins will be calculated bottom to surface

Details

Image volume calculations can change based on optical setting of VPR as well as autodeck setting used to process images For IML2018051 (S2) image volume was calculated as 108155 mm^3 by seascan (6.6 cubic inches) For COR2019002 S2 image volume was calculated as 83663 mm^3 and S3 image volume was calculated as 366082 mm^3

Author(s)

E. Chisholm

---

ctd_cast *Isolate ascending or descending section of ctd cast*

Description

This is an internal step required to bin data

Usage

```
ctd_cast(
  data,
  cast_direction = "ascending",
  data_type,
  cutoff = 0.1,
  breaks = NULL
)
```

Arguments

data	an oce ctd object
cast_direction	'ascending' or 'descending' depending on desired section
data_type	specify 'oce' or 'df' depending on class of desired output
cutoff	Argument passed to ctdFindProfiles
breaks	Argument passed to ctdFindProfiles

Value

Outputs either data frame or oce ctd object

Note

ctdFindProfiles arguments for minLength and cutOff were updated to prevent losing data (EC 2019/07/23)

Author(s)

K Sorochan, E Chisholm

---

ctd_dat_combine *VPR CTD data*

Description

A dataframe including all CTD parameters from the VPR CTD, produced by vpr_ctd_read

Usage

```
ctd_dat_combine
```

Format

A dataframe with 15 variables

time_ms Time stamp when ROI was collected (milliseconds)

conductivity Conductivity collected by the VPR CTD

pressure Pressure measured from the VPR CTD in decibars

temperature Temperature measured from the VPR CTD in celsius

salinity Salinity measured from the VPR CTD

fluor_ref A reference fluorescence baseline provided in millivolts by the VPR CTD for calibrating fluorescence_mv data

fluorescence_mv Fluorescence in millivolts from the VPR CTD (uncalibrated)

turbidity_ref A reference turbidity baseline provided in millivolts for calibrating turbidity_mv

turbidity_mv Turbidity in millivolts from the VPR CTD (uncalibrated)

altitude_NA Altitude data from the VPR CTD

day Day on which VPR data was collected (from AutoDeck)

hour Hour during which VPR data was collected (from AutoDeck)

station Station idnetifier provided during processing

sigmaT Density caluclated from temperature, pressure and salinity data

depth Depth in metres caluclated form pressure

`ctd_df_cols` *Read CTD data (SBE49) and Fluorometer data from CTD- VPR package*

Description

Internal use `vpr_ctd_read`

Usage

```
ctd_df_cols(x, col_list)
```

Arguments

`x`	full filename (ctd .dat file)
`col_list`	list of CTD data column names

Details

WARNING This is hard coded to accept a specific order of CTD data columns. The names and values in these columns can change based on the specific instrument and should be updated before processing data from a new VPR.

Text file format .dat file Outputs ctd dataframe with variables time_ms, conductivity, temperature, pressure, salinity, fluor_ref, fluorescence_mv, turbidity_ref, turbidity_mv, altitude_NA

Author(s)

K. Sorochan, E. Chisholm

`ctd_roi_merge` *VPR CTD data combined with tabulated ROIs*

Description

A dataframe representing CTD data which has been merged with tabulated ROIs in each category, produced by `vpr_ctdroi_merge`

Usage

```
ctd_roi_merge
```

Format

A dataframe with 28 variables

time_ms Time stamp when ROI was collected (milliseconds)

conductivity Conductivity collected by the VPR CTD

pressure Pressure measured from the VPR CTD in decibars

temperature Temperature measured from the VPR CTD in celsius

salinity Salinity measured from the VPR CTD

fluor_ref A reference fluorescence baseline provided in millivolts by the VPR CTD for calibrating fluorescence_mv data

fluorescence_mv Fluorescence in millivolts from the VPR CTD (uncalibrated)

turbidity_ref A reference turbidity baseline provided in millivolts for calibrating turbidity_mv

turbidity_mv Turbidity in millivolts from the VPR CTD (uncalibrated)

altitude_NA Altitude data from the VPR CTD

day Day on which VPR data was collected (from AutoDeck)

hour Hour during which VPR data was collected (from AutoDeck)

station Station identifier provided during processing

sigmaT Density caluclated from temperature, pressure and salinity data

depth Depth in metres caluclated form pressure

roi ROI identification number

categories For each category name (eg. bad_image_blurry, Calanus, krill), there is a line in the dataframe representing the number of ROIs identified in this category

n_roi_total Total number of ROIs in all categories for each CTD data point

ctd_roi_oce *VPR data including CTD and ROI information*

Description

An oce formatted CTD object with VPR CTD and ROI data from package example data set.

Usage

```
ctd_roi_oce
```

Format

An oce package format, a 'CTD' object with VPR CTD and ROI data (1000 data rows)

getRoiMeasurements *THIS FUNCTION HAS BEEN DEPRECATED*

Description

pull roi measurements from all taxa, all files

Usage

```
getRoiMeasurements(taxafolder, nchar_folder, unit = "mm", opticalSetting)
```

Arguments

taxafolder	path to taxa folder (base – autoid folder)
nchar_folder	number of characters in basepath
unit	unit data will be output in, 'mm' (default – millimetres) or 'px' (pixels)
opticalSetting	VPR optical setting determining conversion between pixels and millimetres (options are 'S0', 'S1', 'S2', or 'S3')

Note

This function is very finicky, easily broken because it relies on character string splitting. taxaFolder argument should not end in a backslash, please check output carefully to ensure taxa names or ROI numbers have been properly sub string'd

`insertRow` *INTERNAL USE ONLY quick data frame function from github to insert row inside dat frame*

Description

INTERNAL USE ONLY quick data frame function from github to insert row inside dat frame

Usage

```
insertRow(existingDF, newrow, r)
```

Arguments

`existingDF`	data frame
`newrow`	new row of data
`r`	index of new row

`isopycnal_calculate` *Get vector to draw isopycnal lines on TS plot Used internally to create TS plots*

Description

Get vector to draw isopycnal lines on TS plot Used internally to create TS plots

Usage

```
isopycnal_calculate(sal, pot.temp, reference.p = 0)
```

Arguments

`sal`	salinity vector
`pot.temp`	temperature vector in deg C
`reference.p`	reference pressure for calculation, set to 0

Note

: modified from source:https://github.com/Davidatlarge/ggTS/blob/master/ggTS_DK.R

Author(s)

E. Chisholm

normalize_matrix *Normalize a matrix*

Description

take each element of matrix dived by column total

Usage

```
normalize_matrix(mat)
```

Arguments

mat	a matrix to normalize

Details

Make sure to remove total rows before using with VP data

Note

used internally for visualization of confusion matrices

px_to_mm *Get conversion factor for pixels to mm for roi measurements*

Description

Used internally

Usage

```
px_to_mm(x, opticalSetting)
```

Arguments

x	an aidmea data frame (standard) to be converted into mm from pixels
opticalSetting	the VPR setting determining the field of view and conversion factor between mm and pixels

Details

converts pixels to mm using conversion factor specific to optical setting

Options for opticalSetting are 'S0', 'S1', 'S2', or 'S3'

roimeas_dat_combine *VPR measurement data calculated by Visual Plankton*

Description

A data frame of measurement information for each ROI in the sample data set including long axis length, perimeter and area, produced by vpr_autoid_read

Usage

```
roimeas_dat_combine
```

Format

A data frame with 12 variables

roi Unique ROI identifier - 10 digit

taxa Category in which ROI has been classified by Visual Plankton

day_hour day and hour in which data was collected (from Autodeck)

Perimeter The perimeter of the ROI in millimeters

Area The area of the ROI in millimeters

width1 Width at a first point of the ROI in millimetres (defined in more detail in VPR manual)

width2 Width at a second point of the ROI in millimetres (defined in more detail in VPR manual)

width3 Width at a third point of the ROI in millimetres (defined in more detail in VPR manual)

short_axis_length The length in millimeters of the ROI along the shorter axis

long_axis_length The length in millimeters of the ROI along the longer axis

station Station identifier provided in processing

time_ms Time stamp when ROI was collected in milliseconds

roi_dat_combine *VPR ROI data*

Description

A dataframe including VPR ROI data from the sample dataset, produced by vpr_autoid_read

Usage

```
roi_dat_combine
```

Format

A dataframe with 13 variables

roi Unique ROI identifier - 8 digit

categories For each category name (eg. bad_image_blurry, Calanus, krill), there is a line in the dataframe representing the number of ROIs identified in this category

time_ms Time stamp when ROI was collected (milliseconds)

---

size_df_f *VPR size information dataframe*

Description

A sample data frame of size information from Visual Plankton outputs, processed using `vpr_ctdroisize_merge`

Usage

```
size_df_f
```

Format

An object of class `data.frame` with 14 rows and 14 columns.

Details

@format A dataframe with 14 variables including

frame_ID Unique identifier for each VPR frame

pressure Pressure measured from the VPR CTD in decibars

temperature Temperature measured from the VPR CTD in celsius

salinity Salinity measured from the VPR CTD

sigmaT Density calculated from temperature, salinity and pressure

fluorescence_mv Fluorescence measured by the VPR CTD in millivolts (uncalibrated)

turbidity_mv Turbidity measured by the VPR CTD in millivolts (uncalibrated)

roi Unique ROI identification number - 10 digits, 8 digit millisecond time stamp and two unique digits to denote multiple ROIs within a millisecond

taxa Category in which ROI has been classified by Visual Plankton

day_hour Day and hour in which data was collected, from AutoDeck processing

long_axis_length The length of the longest axis of the ROI image, measured by Visual Plankton

station Station identifier provided during processing

time_ms Time stamp when ROI was collected (milliseconds)

roi_ID ROI identification number- 8 digit time stamp, without unique 2 digit ending

taxa_conc_n *A binned data frame of concentration data per category*

Description

A 'binned' dataframe from sample VPR data, including concentrations of each category, where each data point represents a 5 metre bin of averaged VPR data. Produced using vpr_roi_concentration

Usage

```
taxa_conc_n
```

Format

A dataframe with 21 variables

depth Depth calculated from pressure in metres

min_depth The minimum depth of the bin in metres

max_depth The maximum depth of the bin in metres

depth_diff The difference between minimum and maximum bin depth in metres

min_time_s The minimum time in seconds of the bin

max_time_s The maximum time in seconds of the bin

time_diff_s The difference between minimum and maximum time in a bin, in seconds

n_roi_bin The number of ROI observations in a bin

conc_m3 The concentration of ROIs in a bin, calculated based on image volume and number of frames per bin

temperature Temperature measured from the VPR CTD in celsius (averaged within the bin)

salinity Salinity measured from the VPR CTD (averaged within the bin)

density sigma T density calculated from temperature, salinity and pressure (averaged within the bin)

fluorescence Fluorescence measured by the VPR CTD in millivolts (uncalibrated) (averaged within the bin)

turbidity Turbidity measured by the VPR CTD in millivolts (uncalibrated) (averaged within the bin)

avg_hr The mean time in which bin data was collected, in hours

n_frames The number of frames captured within a bin

vol_sampled_bin_m3 The volume of the bin sampled in metres cubed

toyo Identifier of the tow-yo section which bin is a part of, either ascending or descending, appended by a number

max_cast_depth The maximum depth of the entire VPR cast

taxa The category in which ROIs in bin have been classified by Visual Plankton

station Station identifier provided during processing

vpr_autoid_check *Checks manually created aid files for errors*

Description

Removes any empty aid files after manual reclassification, checks for tow numbers and other metadata to match. Performs check to ensure measurement and ROI files are the same length

Usage

```
vpr_autoid_check(basepath, cruise, del)
```

Arguments

basepath	basepath to autoid folder eg. C:/data/CRUISENAME/autoid/
cruise	name of cruise which is being checked
del	Logical value, if TRUE, empty files will be deleted (see warning), if FALSE, files WILL NOT be deleted (they will be listed in output)

Details

WARNING: This function will delete empty aid and aidmeas files, permanently changing your directory. Consider making a back up copy before running this function.

Value

text file (saved in working directory) named CRUISENAME_aid_file_check.txt

Author(s)

E Chisholm

vpr_autoid_copy *Copy VPR images into folders*

Description

Organize VPR images into folders based on classifications provided by visual plankton

Usage

```
vpr_autoid_copy(basepath, day, hour, classifier_type, classifier_name, taxa)
```

Arguments

basepath	A file path to your autoid folder where VP data is stored eg. "C:\\data\\cruise_XXXXXXXXXXXXXXX\\
day	character string representing numeric day of interest
hour	character string representing hour of interest
classifier_type	character string representing the type of classifier (either 'svm', 'nn' or 'dual') from Visual Plankton
classifier_name	character string representing name of Visual Plankton classifier
taxa	optional list of character strings if you wish to only copy images from specific classification groups

Value

organized file directory where VPR images are contained with folders, organized by day, hour and classification, inside your basepath/autoid folder

---

vpr_autoid_create *Modifies aid and aid mea files based on manual reclassification*

Description

Modifies aid and aid mea files based on manual reclassification

Usage

```
vpr_autoid_create(reclassify, misclassified, basepath, day, hour, mea = TRUE)
```

Arguments

reclassify	list of reclassify files (output from vpr_manual_classification())
misclassified	list misclassify files (output from vpr_manual_classification())
basepath	base path to auto ID folder eg 'E:/autoID_EC_07032019/'
day	day identifier for relevant aid & aidmeas files
hour	hour identifier for relevant aid & aidmeas files
mea	logical indicating whether or not there are accompanying measurement files to be created

examples:

basepath <- 'E:/autoID_EC_07032019/' day <- '289' hr <- '08' day_hour_files <- paste0('d', day, '.h', hr) misclassified <- list.files(day_hour_files, pattern = 'misclassified_', full.names = TRUE) reclassify <- list.files(day_hour_files, pattern = 'reclassify_', full.names = TRUE) vpr_autoid_create(reclassify, misclassified, basepath)

Author(s)

E. Chisholm

vpr_autoid_read *Read VPR aid files*

Description

Read aid text files containing ROI string information or measurement data and output as a dataframe

Usage

```
vpr_autoid_read(
  file_list_aid,
  file_list_aidmeas,
  export,
  station_of_interest,
  opticalSetting,
  warn = TRUE
)
```

Arguments

file_list_aid	a list object of aid text files, containing roi strings. Output from matlab Visual Plankton software.
file_list_aidmeas	a list object of aidmea text files, containing ROI measurements. Output from matlab Visual Plankton software.
export	a character string specifying which type of data to output, either 'aid' (roi strings) or 'aidmeas' (measurement data)
station_of_interest	Station information to be added to ROI data output, use NA if irrelevant
opticalSetting	Optional argument specifying VPR optical setting. If provided will be used to convert size data into mm from pixels, if missing size data will be output in pixels
warn	Logical, FALSE silences size data unit warnings

Details

Only outputs either ROI string information OR measurement data

Note

Full paths to each file should be specified

Author(s)

E. Chisholm & K. Sorochan

Examples

```
station_of_interest <- 'test'
dayhour <- c('d222.h03', 'd222.h04')

#' #VPR OPTICAL SETTING (S0, S1, S2 OR S3)
opticalSetting <- "S2"
imageVolume <- 83663 #mm^3

auto_id_folder <- system.file('extdata/COR2019002/autoid/', package = 'vprr', mustWork = TRUE)
auto_id_path <- list.files(paste0(auto_id_folder, "/"), full.names = TRUE)

#'   # Path to aid for each taxa
aid_path <- paste0(auto_id_path, '/aid/')
# Path to mea for each taxa
aidmea_path <- paste0(auto_id_path, '/aidmea/')

# AUTO ID FILES
aid_file_list <- list()
aidmea_file_list <- list()
for (i in 1:length(dayhour)) {
  aid_file_list[[i]] <-
    list.files(aid_path, pattern = dayhour[[i]], full.names = TRUE)
  # SIZE DATA FILES
  aidmea_file_list[[i]] <-
    list.files(aidmea_path, pattern = dayhour[[i]], full.names = TRUE)
}

aid_file_list_all <- unlist(aid_file_list)
aidmea_file_list_all <- unlist(aidmea_file_list)

 # ROIs
roi_dat_combine <-
  vpr_autoid_read(
    file_list_aid = aid_file_list_all,
    file_list_aidmeas = aidmea_file_list_all,
    export = 'aid',
    station_of_interest = station_of_interest,
    opticalSetting = opticalSetting,
    warn = FALSE
  )

# MEASUREMENTS
roimeas_dat_combine <-
  vpr_autoid_read(
    file_list_aid = aid_file_list_all,
    file_list_aidmeas = aidmea_file_list_all,
    export = 'aidmeas',
    station_of_interest = station_of_interest,
    opticalSetting = opticalSetting,
    warn = FALSE
 )
```

vpr_category *Get taxa ids from string*

Description

Get taxa ids from string

Usage

```
vpr_category(x)
```

Arguments

x	A string specifying the directory of the "taxafolder", containing the taxa id

Value

A string of only the taxa id

Author(s)

K Sorochan

See Also

vpr_hour, vpr_day, vpr_roi

Examples

```
taxa_string <- 'C:/data/cruise/autoid/Calanus/d000/h00'
vpr_category(taxa_string)
```

vpr_category_create *Create a new taxa to be considered for classification after processing with VP*

Description

creates empty directory structure to allow consideration of new taxa during vpr_manual_classification()

Usage

```
vpr_category_create(taxa, basepath)
```

Arguments

taxa	new taxa name to be added (can be a list of multiple taxa names)
basepath	basepath used for vpr_manual_classification

Value

empty directory structure using new taxa name inside basepath

vpr_ctdroisize_merge *Format CTD and Size data from VPR*

Description

Format CTD and Meas data frames into combined data frame for analysis and plotting of size data

Usage

```
vpr_ctdroisize_merge(data, data_mea, taxa_of_interest)
```

Arguments

data	VPR dataframe from vpr_ctdroi_merge, with calculated variable sigmaT
data_mea	VPR size data frame from vpr_autoid_read
taxa_of_interest	a list of taxa of interest to be included in output dataframe

Value

A dataframe containing VPR CTD and size data

Examples

```
data("ctd_roi_merge")
data("roimeas_dat_combine")
category_of_interest = 'Calanus'

ctd_roi_merge$time_hr <- ctd_roi_merge$time_ms /3.6e+06

size_df_f <- vpr_ctdroisize_merge(ctd_roi_merge, data_mea = roimeas_dat_combine,
 taxa_of_interest = category_of_interest)
```

---

vpr_ctdroi_merge *Merge CTD and ROI data from VPR*

Description

Combines CTD data (time, hydrographic parameters), with ROI information (identification number) into single dataframe, aligning ROI identification numbers and taxa classifications with time and hydrographic parameters

Usage

```
vpr_ctdroi_merge(ctd_dat_combine, roi_dat_combine)
```

Arguments

ctd_dat_combine	a CTD dataframe from VPR processing from vpr_ctd_read
roi_dat_combine	a data frame of roi aid data from vpr_autoid_read

Author(s)

E. Chisholm & K. Sorochan

Examples

```
data('ctd_dat_combine')
data('roi_dat_combine')

ctd_roi_merge <- vpr_ctdroi_merge(ctd_dat_combine, roi_dat_combine)
```

vpr_ctd_files *Create a list of ctd files to be read*

Description

Searches through typical VP directory structure

Usage

```
vpr_ctd_files(castdir, cruise, day_hour)
```

Arguments

castdir	root directory for ctd cast files
cruise	cruise name (exactly as in directory structure)
day_hour	vector of day-hour combinations (e.g, dXXX.hXX)

Details

Use with caution

Value

vector of ctd file paths matching days-hour combinations provided

Author(s)

E. Chisholm and K. Sorochan

vpr_ctd_read *Read and format CTD VPR data*

Description

Acts as a wrapper for ctd_df_cols

Usage

```
vpr_ctd_read(ctd_files, station_of_interest, day, hour, col_list)
```

Arguments

ctd_files	full file paths to vpr ctd .dat files
station_of_interest	VPR station name
day	Day of interest, if not provided will be pulled from file path
hour	Hour of interest, if not provided will be pulled from file path
col_list	Optional list of CTD data column names

Details

Reads CTD data and adds day, hour, and station information. Calculates sigma T and depth variables from existing CTD data to supplement raw data. If there are multiple hours of CTD data, combines them into single dataframe.

WARNING ctd_df_cols is hard coded to accept a specific order of CTD data columns. The names and values in these columns can change based on the specific instrument and should be updated/confirmed before processing data from a new VPR.

Author(s)

E. Chisholm & K. Sorochan

Examples

```
station_of_interest <- 'test'

ctd_files <- system.file("extdata/COR2019002/rois/vpr5/d222", "h03ctd.dat",
package = "vprr", mustWork = TRUE)

ctd_dat_combine <- vpr_ctd_read(ctd_files, station_of_interest)
```

---

vpr_ctd_ymd *Add Year/ month/ day hour:minute:second information*

Description

Calculate and record calendar dates for vpr data from day-of-year, hour, and time (in milliseconds) info. Will also add 'time_hr' parameter if not already present.

Usage

```
vpr_ctd_ymd(data, year, offset)
```

Arguments

data	VPR data frame from vpr_ctdroi_merge
year	Year of data collection
offset	time offset in hours between VPR CPU and processed data times (optional)

Value

a VPR data frame with complete date/time information in a new row named 'ymdhms'

Examples

```
year <- 2019
data('ctd_roi_merge')
dat <- vpr_ctd_ymd(ctd_roi_merge, year)
```

---

vpr_day *Get day identifier*

Description

Get day identifier

Usage

```
vpr_day(x)
```

Arguments

x	A string specifying the directory and file name of the size file

Value

A string of only the day identifier (i.e., "dXXX")

Author(s)

K Sorochan

See Also

vpr_hour, vpr_roi, vpr_category

Examples

```
day_string <- 'C:/data/cruise/autoid/Calanus/d000/h00'
vpr_day(day_string)
```

vpr_dayhour — *Find day & hour info to match each station of interest for processing*

Description

@author E. Chisholm and K. Sorochan

Usage

```
vpr_dayhour(stations, file)
```

Arguments

stations	a vector of character values naming stations of interest
file	CSV file containing 'day', 'hour', 'station', and 'day_hour' columns

Value

Vector of day-hour combinations corresponding to stations of interest

vpr_hour — *Get hour identifier*

Description

Get hour identifier

Usage

```
vpr_hour(x)
```

Arguments

x	A string specifying the directory and file name of the size file

Value

A string of only the hour identifier (i.e., "hXX")

Author(s)

K Sorochan

See Also

vpr_day, vpr_roi, vpr_category

Examples

```
hour_string <- 'C:/data/cruise/autoid/Calanus/d000/h00'
vpr_hour(hour_string)
```

`vpr_img_category` *Explore images by depth and classification*

Description

Pulls images from specific depth ranges in specific classification group

Usage

```
vpr_img_category(
  data,
  min.depth,
  max.depth,
  roiFolder,
  format = "list",
  taxa_of_interest
)
```

Arguments

`data`	data frame containing CTD and ROI data from `vpr_ctdroi_merge`, which also contains calculated variables sigmaT and time_hr
`min.depth`	minimum depth of ROIs you are interested in looking at
`max.depth`	maximum depth of ROIs you are interested in exploring
`roiFolder`	directory that ROIs are within (can be very general eg. C:/data, but will be quicker to process with more specific file path)
`format`	option of how images will be output, either as 'list' a list of file names or 'image' where images will be displayed
`taxa_of_interest`	character string of classification group from which to pull images

vpr_img_check *Remove ROI strings from aid and aidmeas files based on a manually organized folder of images*

Description

Should be used after vpr_img_copy, and manual image removal from created folders

Usage

```
vpr_img_check(folder_dir, basepath)
```

Arguments

folder_dir	directory path to day hour folders containing manually reorganized images of a specific taxa eg. 'C:/data/cruise_IML2018051/krill/images/' where that folder contains '.....d123.h01/' which contains manually sorted images of krill
basepath	directory path to original Visual Plankton files, specified down to the classification group. eg. 'C:/data/cruise_IML2018051/autoid/krill'

vpr_img_copy *Image copying function for specific taxa of interest*

Description

This function can be used to copy images from a particular taxa, day and hour into distinct folders within the auto id directory This is useful for visualizing the ROIs of a particular classification group or for performing manual tertiary checks to remove images not matching classification group descriptions.

Usage

```
vpr_img_copy(auto_id_folder, taxas.of.interest, day, hour)
```

Arguments

auto_id_folder	eg "D:/VP_data/IML2018051/autoid"
taxas.of.interest	eg. taxas.of.interest <- c('Calanus')
day	character, day of interest
hour	character, hour of interest

vpr_img_depth *Explore VPR images by depth bin*

Description

Allows user to pull VPR images from specific depth ranges, to investigate trends before classification of images into taxa groups

Usage

```
vpr_img_depth(data, min.depth, max.depth, roiFolder, format = "list")
```

Arguments

data	data frame containing CTD and ROI data from vpr_ctdroi_merge, which also contains calculated variables sigmaT and time_hr
min.depth	minimum depth of ROIs you are interested in looking at
max.depth	maximum depth of ROIs you are interested in exploring
roiFolder	directory that ROIs are within (can be very general eg. C:/data, but will be quicker to process with more specific file path)
format	option of how images will be output, either as 'list' a list of file names or 'image' where images will be displayed

vpr_img_reclassified *Explore reclassified images*

Description

Pull image from reclassified or misclassified files produced during vpr_manual_classification

Usage

```
vpr_img_reclassified(day, hour, base_dir, taxa_of_interest, image_dir)
```

Arguments

day	Character string, 3 digit day of interest of VPR data
hour	Character string, 2 digit hour of interest of VPR data
base_dir	directory path to folder containing day/hour folders in which misclassified and reclassified files are organized (eg.'C:/VPR_PROJECT/r_project_data_vis/classification files/') which would contain 'd123.h01/reclassified_krill.txt')
taxa_of_interest	Classification group from which to pull images
image_dir	directory path to ROI images, eg. "E:\\data\\cruise_IML2018051\\", file separator MUST BE "\\" in order to be recognized

Value

folders of misclassified or reclassified images inside image_dir

---

`vpr_manual_classification` *Function to check results of classification manually*

Description

Displays each image in day hour specified, prompts user to confirm or deny classification. If classification is denied, asks for a reclassification value based on available taxa

Usage

```
vpr_manual_classification(
  day,
  hour,
  basepath,
  taxa_of_interest,
  gr = TRUE,
  scale = "x300",
  opticalSetting = "S2",
  img_bright = TRUE
)
```

Arguments

`day`	day of interest in autoid
`hour`	hour of interest in autoid
`basepath`	file path to auto id folder eg 'E:/autoID_EC_07032019/'
`taxa_of_interest`	list of taxa folders you wish you sort through
`gr`	logical indicating whether pop up graphic menus are used (user preference - defaults to TRUE)
`scale`	argument passed to `image_scale`, default = 'x300'
`opticalSetting`	specifies optical setting of VPR, defining image frame size, current options are 'S0', 'S1', 'S2' (default), 'S3', see further info in details
`img_bright`	logical value indicating whether or not to include a blown out high brightness version of image (can be helpful for viewing dark field fine appendages)

Details

Optical Setting frame sizes: S0 = 7x7 mm, S1 = 14x14mm, S2 = 24x24mm, S3 = 48x48 mm. These settings define the conversion factor from pixels to millimetres and calculate image size for classification reference

Development

- Add "undo" functionality to go back on a typing mistake
- Fix scaling/ size issue so images are consistently sized
- show ROI number for image somewhere for reference when in doubt of classification

---

vpr_oce_create *Create ctd oce object with vpr data*

Description

Formats VPR data frame into oce format CTD object

Usage

```
vpr_oce_create(data)
```

Arguments

data	data frame of vpr data with variable names 'time_ms', 'fluorescence_mv', 'turbidity_mv', 'n_roi', 'sigmaT'

Author(s)

E. Chisholm

Examples

```
data('ctd_roi_merge')
oce_dat <- vpr_oce_create(ctd_roi_merge)
```

---

vpr_plot_contour *Interpolated contour plot of particular variable*

Description

Creates interpolated contour plot, can be used as a background for ROI or tow yo information

Usage

```
vpr_plot_contour(
  data,
  var,
  dup = "mean",
  method = "interp",
  labels = TRUE,
  bw = 1,
  cmo
)
```

Arguments

data	data frame needs to include time_hr, depth, and variable of choice (var)
var	variable in dataframe which will be interpolated and plotted
dup	if method == 'interp'. Method of handling duplicates in interpolation, passed to interp function (options: 'mean', 'strip', 'error')
method	Specifies interpolation method, options are 'interp' or 'oce', oce uses slightly different method (oce is least error prone)
labels	logical value indicating whether or not to plot contour labels
bw	bin width defining interval at which contours are labelled
cmo	name of a cmocean plotting theme, see ?cmocean for more information

Author(s)

E. Chisholm & Kevin Sorochan

---

vpr_plot_histsize *Plot size frequency histogram*

Description

Plot size frequency histogram

Usage

```
vpr_plot_histsize(data, param, title = NULL, bw = 0.1, xlim = NULL)
```

Arguments

data	size data from auto_measure_mm subset into taxas
param	size parameter of interest (corresponds to sub lists within data argument)
title	main title for plot, if left null will default based on parameter and taxa
bw	bin width, defines width of bars on histogram, defaults to 0.1, decrease for more detail
xlim	plot xlimit, defaults to min max of data if not provided

Note

param options are typically 'Perimeter', 'Area', 'width1','width2', 'width3', 'short_axis_length', 'long_axis_length'

Author(s)

E. Chisholm

`vpr_plot_profile` *Plots VPR profiles of temperature, salinity, density, fluorescence and concentration (by classification group)*

Description

This plot allows a good overview of vertical distribution of individual classification groups along with reference to hydrographic parameters. Facet wrap is used to create distinct panels for each taxa provided

Usage

```
vpr_plot_profile(taxa_conc_n, taxa_to_plot, plot_conc)
```

Arguments

`taxa_conc_n`	A VPR data frame with hydrographic and concentration data separated by taxa (from `vpr_roi_concentration`)
`taxa_to_plot`	The specific classification groups which will be plotted, if NULL, will plot all taxa combined
`plot_conc`	Logical value whether or not to include a concentration plot (FALSE just shows CTD data)

Value

A gridded object of at least 3 ggplot objects

vpr_plot_sizefreq *Size Frequency plots for VPR data*

Description

This uses the hist plot function in base R to give a histogram of size (long axis length) frequency within a taxa. **!!WARNING:** this function uses hard coded plot attributes

Usage

```
vpr_plot_sizefreq(x, number_of_classes, colour_of_bar)
```

Arguments

x	a data frame with columns 'taxa', 'long_axis_length'
number_of_classes	numeric value passed to nclass argument in hist()
colour_of_bar	character value defining colour of plotted bars

Author(s)

K. Sorochan

vpr_plot_TS *Make a balloon plot against a TS plot*

Description

TS balloon plot with ROI concentration, sorted by taxa includes isopycnal line calculations

Usage

```
vpr_plot_TS(x, reference.p = 0, var)
```

Arguments

x	dataframe with temperature, salinity, number of rois (n_roi_bin)
reference.p	reference pressure (default at 0 for surface)- used to calculate isopycnals
var	variable on which size of points will be based, eg conc_m3 or n_roi_bin

Note

modified from source: https://github.com/Davidatlarge/ggTS/blob/master/ggTS_DK.R

Author(s)

E. Chisholm

vpr_plot_TScat *Make a balloon plot*

Description

Balloon plot against a TS plot with ROI concentration and sorted by taxa includes isopycnal line calculations. Version of vpr_plot_TS, with only relevant* taxa specified. *to current analysis and research objectives (See note).

Usage

```
vpr_plot_TScat(x, reference.p = 0)
```

Arguments

x	dataframe with temperature, salinity, number of rois named by taxa
reference.p	reference pressure (default at 0 for surface)- used to calculate isopycnals

Note

WARNING HARD CODED FOR 5 TAXA, CALANUS, KRILL, ECHINODERM LARVAE, SMALL COPEPOD, CHAETOGNATHS !! Uses isopycnal labelling method which does not label every contour

modified from source: https://github.com/Davidatlarge/ggTS/blob/master/ggTS_DK.R

vpr_pred_read *Read prediction output from a CNN model*

Description

Read prediction output from a CNN model

Usage

```
vpr_pred_read(filename)
```

Arguments

filename	model prediction output file (.txt) from vpr_transferlearn::save_output()

Value

a dataframe

vpr_roi *Get roi ids from string*

Description

Get roi ids from string

Usage

```
vpr_roi(x)
```

Arguments

x	A string specifying directory and file name of roi

Value

A string of only the 10 digit roi identifier

Author(s)

K Sorochan

See Also

vpr_hour, vpr_day, vpr_category

Examples

```
roi_string <- 'roi.0100000000.tif'
vpr_roi(roi_string)
```

vpr_roi_concentration *Calculate VPR concentrations*

Description

Calculates concentrations for each named taxa in dataframe

Usage

```
vpr_roi_concentration(
  data,
  taxas_list,
  station_of_interest,
  binSize,
  imageVolume
)
```

Arguments

data	a VPR dataframe as produced by vpr_ctdroi_merge
taxas_list	a list of character strings representing taxa present in the station being processed
station_of_interest	The station being processed
binSize	passed to bin_calculate, determines size of depth bins over which data is averaged
imageVolume	the volume of VPR images used for calculating concentrations (mm^3)

Examples

```
data('ctd_roi_merge')
ctd_roi_merge$time_hr <- ctd_roi_merge$time_ms /3.6e+06

taxas_list <- c('Calanus', 'krill')
binSize <- 5
station_of_interest <- 'test'
imageVolume <- 83663

taxa_conc_n <- vpr_roi_concentration(ctd_roi_merge, taxas_list,
station_of_interest, binSize, imageVolume)
```

---

vpr_save *Save VPR data as an as.oce object*

Description

Save VPR data as an as.oce object

Usage

```
vpr_save(data, metadata)
```

Arguments

data	a VPR data frame
metadata	(optional) a named list of character values giving metadata values. If this argument is not provided user will be prompted for a few generic metadata requirements.

Details

This function will pass a VPR data frame object to an oce object. Using an oce object as the default export format for VPR data allows for metadata and data to be kept in the same, space efficient file, and avoid redundancy in the data frame. The function check for data parameters that may actually be metadata parameters (rows which have the same value repeated for every observation). These parameters will automatically be copied into the metadata slot of the oce object. The function will also prompt for a variety of required metadata fields. Depending on specific research / archiving requirements, these metadata parameters could be updated by providing the argument metadata.

Default metadata parameters include 'deploymentType', 'waterDepth', 'serialNumber', 'latitude', 'longitude', 'castDate', 'castStartTime', 'castEndTime', 'processedBy', 'opticalSetting', 'imageVolume', 'comment'.

Value

an oce CTD object with all VPR data as well as metadata

Examples

```
data("taxa_conc_n")
metadata <- c('deploymentType' = 'towyo', 'waterDepth' =
max(ctd_roi_merge$pressure), 'serialNumber' = NA, 'latitude' = 47,
'longitude' = -65, 'castDate' = '2019-08-11', 'castStartTime'= '00:00',
'castEndTime' = '01:00', 'processedBy' = 'E. Chisholm', 'opticalSetting' =
'S2', 'imageVolume' = 83663, 'comment' = 'test data')

oce_dat <- vpr_save(taxa_conc_n, metadata)
# save(oce_dat, file = vpr_save.RData') # save data
```

---

vpr_size_bin *Bin VPR size data*

Description

Calculates statistics for VPR measurement data in depth averaged bins for analysis and visualization

Usage

```
vpr_size_bin(data_all, bin_mea)
```

Arguments

data_all	a VPR CTD and measurement dataframe from vpr_ctdroisize_merge
bin_mea	Numerical value representing size of depth bins over which data will be combined, unit is metres, typical values range from 1 - 5

Value

a dataframe of binned VPR size data statistics including number of observations, median, interquartile ranges, salinity and pressure, useful for making boxplots

Examples

```
data('size_df_f')
vpr_size_bin(size_df_f, bin_mea = 5)
```

---

vpr_summary *Data Summary Report*

Description

Part of VP easy plot processing, prints data summary report to give quantitative, exploratory analysis of data

Usage

```
vpr_summary(all_dat, fn, tow = tow, day = day, hour = hour)
```

Arguments

all_dat	data frame containing VPR and CTD data including time_ms, time_hr, conductivity, temperature, pressure, salinity, fluorescence_mv, turbidity_mv, sigmaT
fn	file name to save data summary, if not provided, summary will print to console
tow	VPR tow number
day	julian day
hour	two digit hour (24 hr clock)

Author(s)

E Chisholm

vpr_trrois_size *Get size data from idsize files*

Description

useful for getting size distribution of known rois from each taxa. gathers size information from idsize text files produced when training a new classifier in VP (Visual Plankton)

Usage

```
vpr_trrois_size(directory, taxa, opticalSetting)
```

Arguments

directory	cruise directory eg. 'C:/data/IML2018051/'
taxa	list of character elements containing taxa of interest
opticalSetting	VPR optical setting determining conversion between pixels and millimetres (options are 'S0', 'S1', 'S2', or 'S3')

vp_plot_matrix *Plots normalized confusion matrix*

Description

Plots normalized confusion matrix

Usage

```
vp_plot_matrix(cm, classes, type, addLabels = TRUE, threshold = NULL)
```

Arguments

cm	Confusion matrix (numeric)
classes	character list of classes present in confusion matrix (ordered)
type	character value 'NN', 'SVM' or 'Dual', appended to 'Confusion Matrix' to create title
addLabels	logical value whether to add percentage accuracy labels to plot (defaults to TRUE)
threshold	numeric value which determines the minimum value of frequency labelled on the plot on a normalized scale of 0-1 (useful for highlighting significant disagreement)

Value

a visualization of the confusion matrix, normalized

Author(s)

E. Chisholm

`vp_plot_unkn` *Function to visualize losses to unknown category due to disagreement in Dual classifier*

Description

Makes confusion matrix like plot, where x axis represent SVM classification, y axis represent NN classification Allows visual summary of data lost to unknown category

Usage

```
vp_plot_unkn(cm, classes, threshold = 0, summary = TRUE, sample_size = NULL)
```

Arguments

`cm`	dual unknown confusion matrix from VP
`classes`	taxa groups in order, from VP
`threshold`	minimum value which will be labelled in plot
`summary`	logical to add text summary to plot E. Chisholm May 2019
`sample_size`	character string describes the sample size used to train the model being plotted (optional)

Index

∗ **datasets**
ctd_dat_combine, 6
ctd_roi_merge, 8
ctd_roi_oce, 9
roi_dat_combine, 12
roimeas_dat_combine, 12
size_df_f, 13
taxa_conc_n, 14

as.oce, *36*

bin_calculate, 3, *4, 5*, *36*
bin_cast, *3, 4*, 4

concentration_category, *4*, 4
ctd_cast, *3, 4*, 5
ctd_dat_combine, 6
ctd_df_cols, 7, *22, 23*
ctd_roi_merge, 8
ctd_roi_oce, 9
ctdFindProfiles, *6*

getRoiMeasurements, 9

hist, *33*

image_scale, *29*
insertRow, 10
isopycnal_calculate, 10

normalize_matrix, 11

px_to_mm, 11

roi_dat_combine, 12
roimeas_dat_combine, 12

size_df_f, 13
swDepth, *3*

taxa_conc_n, 14

vp_plot_matrix, 39
vp_plot_unkn, 40
vpr_autoid_check, 15
vpr_autoid_copy, 15
vpr_autoid_create, 16
vpr_autoid_read, *12*, 17, *20, 21*
vpr_category, 19, *24, 25*, *35*
vpr_category_create, 20
vpr_ctd_files, 22
vpr_ctd_read, *6, 7*, *21*, 22
vpr_ctd_ymd, 23
vpr_ctdroi_merge, *8*, *20*, 21, *24*, *26*, *28*, *36*
vpr_ctdroisize_merge, *13*, 20, *38*
vpr_day, *19*, 24, *25*, *35*
vpr_dayhour, 25
vpr_hour, *19*, *24*, 25, *35*
vpr_img_category, 26
vpr_img_check, 27
vpr_img_copy, *27*, 27
vpr_img_depth, 28
vpr_img_reclassified, 28
vpr_manual_classification, *28*, 29
vpr_oce_create, *4*, 30
vpr_plot_contour, 30
vpr_plot_histsize, 31
vpr_plot_profile, 32
vpr_plot_sizefreq, 33
vpr_plot_TS, 33, *34*
vpr_plot_TScat, 34
vpr_pred_read, 34
vpr_roi, *19*, *24, 25*, 35
vpr_roi_concentration, *4*, *14*, *32*, 35
vpr_save, 36
vpr_size_bin, 37
vpr_summary, 38
vpr_trrois_size, 39