Package ‘Decomp2d’

August 9, 2022

Version 0.6.0

Date 2022-01-20

Title Two-Dimensional Decomposition

Author Donghoh Kim [aut, cre], Hee-Seok Oh [ctb], Guebin Choi [aut]

Maintainer Donghoh Kim <donghoh.kim@gmail.com>

Depends R (>= 3.0), EMD (>= 1.5.9), EPT (>= 0.7.5), imagerExtra (>= 1.3.2), Rssa (>= 1.0.4), wavethresh (>= 4.6.8)

Description Two-dimensional decomposition for an image is implemented.

License GPL (>= 3)

NeedsCompilation no

R topics documented:

`d11` *D11 texture*

Description

A 128x128 image of D11 texture.

Usage

```
data(d11)
```

Format

A 128x128 matrix.

Examples

```
data(d11)
image(d11, xlab="", ylab="", main="", col=gray(0:100/100), axes=FALSE)
```

decomp2d *Decomposition of an Image*

Description

This function decomposes an image into two components according to frequency.

Usage

```
decomp2d(z, method="wavelet",
    dct.frequency=NULL,
    emd.sm=FALSE, emd.spar=NULL, emd.tol=0.1^2, emd.maxiter=20,
    ept.tau=NULL, ept.process=c("average", "average"),
    ept.tol=0.1^2, ept.maxiter=20,
    pca.freqcomp=NULL,
    ssa.L=NULL, ssa.freqcomp=NULL,
    wavelet.highlevel=NULL)
```

Arguments

z matrix of an image.

method decomposition method of "dct" for discrete cosine transform, "emd" for bidimensional empirical mode decomposition, "ept" for ensemble pactch transform, "pca" for two-dimensional principal component analysis, "ssa" for two-dimensional singular spectrum analysis, and "wavelet" for wavelet transform.

dct.frequency threshold of frequencies for "dct".

emd.sm specifies whether envelop is constructed by interpolation or local polynomial smoothing for "emd". Use FALSE for interpolation or TRUE for local polynomial smoothing.

emd.spar specifies user-supplied smoothing parameter of local polynomial smoothing for "emd".

emd.tol tolerance for stopping rule of sifting for "emd".

emd.maxiter the maximum number of sifting for "emd".

ept.tau a size parameter for "ept": ept.tau[1] for horizontal size and ept.tau[2] for vertical size of a two-dimensional patch. When length(ept.tau) is 1, the horizontal and vertical size are the same.

ept.process specifies transform types for "ept": ept.process[1] for patch process and ept.process[2] for ensemble process.

ept.tol tolerance for stopping rule of sifting for "ept".

ept.maxiter the maximum number of sifting for "ept".

pca.freqcomp numeric vectors of frequency components for "pca".

ssa.L numeric vector with length 2 of window length for "ssa".

ssa.freqcomp numeric vectors of frequency components for "ssa".

wavelet.highlevel
specifies resolution level of high-frequency component for "wavelet".

Details

This function decomposes an image into frequency component and residue of two-dimensional image.

Value

`fc`	high-frequency component decomposed from an image `z`.
`residue`	residue image decomposed from an image `z`.

See Also

`empperiod`.

Examples

```
#### example : composite of two components having different frequencies
nr <- nc <- 128; x <- seq(0, 1, length=nr); y <- seq(0, 1, length=nc)

coscomp1 <- outer(cos(20 * pi * x), cos(20 * pi * y))
coscomp2 <- outer(cos(5* pi * x), cos(5 * pi * y))
cosmeanf <- coscomp1 + coscomp2

op <- par(mfcol=c(3,1), mar=c(0,0.5,2,0.5))
image(cosmeanf, xlab="", ylab="", main="a composite image",
    col=gray(0:100/100), axes=FALSE)
image(coscomp1, xlab="", ylab="", main="high-frequency component",
    col=gray(0:100/100), axes=FALSE)
image(coscomp2, xlab="", ylab="", main="low-frequency component",
    col=gray(0:100/100), axes=FALSE)

#### Decomposition by Wavelet Transform
outcoswr3 <- decomp2d(cosmeanf, method="wavelet", wavelet.highlevel=3)

par(mfcol=c(2,1), mar=rep(0.1, 4), oma=c(0,1.35,1.35,0))
image(outcoswr3$fc, xlab="", ylab="", main="", col=gray(0:100/100), axes=FALSE)
mtext("high-frequency component", side = 2, line = 0.3, cex=0.85, font=2)
mtext("level 3", side=3, line=0.1, cex=0.85, font=2)
image(outcoswr3$residue, xlab="", ylab="", main="", col=gray(0:100/100), axes=FALSE)
mtext("low-frequency component", side=2, line=0.3, cex=0.85, font=2)
par(op)
```

`empperiod` *Calculating Empirical Period of an Image*

Description

This function calculates empirical period of an image.

Usage

```
empperiod(z)
```

Arguments

z	matrix of an input image

Details

This function calculates empirical period of an image.

Value

rowperiod	vector of empirical period between row-wise local maxima.
colperiod	vector of empirical period between column-wise local maxima.

See Also

decomp2d.

Examples

```
nr <- 128; x <- y <- seq(0, 1, length=nr)

coscomp1 <- outer(cos(20 * pi * x), cos(20 * pi * y))
coscomp2 <- outer(cos(5* pi * x), cos(5 * pi * y))
cosmeanf <- coscomp1 + coscomp2

op <- par(mfrow=c(1,2), mar=c(2,2,1,1))
hist(empperiod(cosmeanf)$rowperiod, xaxt = "n", breaks=seq(4, 55, by=3), freq=FALSE,
    main="empirical period of vertical direction", xlab="")
axis(1, seq(4, 55, by=3), seq(4, 55, by=3))
hist(empperiod(cosmeanf)$colperiod, xaxt = "n", breaks=seq(4, 55, by=3), freq=FALSE,
    main="empirical period of horizontal direction", xlab="")
axis(1, seq(4, 55, by=3), seq(4, 55, by=3))
par(op)
```

Index