

# Package: SatRbedo (via r-universe)

June 7, 2026

**Title** Tools for Retrieving Snow and Ice Albedo from Optical Satellite Imagery

**Version** 1.0.0

**Description** The package consists of a set of tools for retrieving snow and ice albedo from optical satellite imagery (e.g., Landsat and Sentinel-2). The tools require the following gridded datasets: (1) atmospherically-corrected surface reflectance, (2) a digital elevation model (DEM), and (3) satellite and solar azimuth and zenith angles. The package includes tools for: image pre-processing (crop grids to a specified extent, project grids with different coordinate systems, and convert data from integer to floating point); convert nadir satellite observations to off-nadir values using view-angle corrections; detect and mask topographic shadows; automatic discrimination of snow and ice surfaces; anisotropic correction of reflected radiation of glacier snow and ice using empirical parameterizations of the bidirectional reflectance distribution function (BRDF); topographic correction; and narrow-to-broadband albedo conversion.

**License** GPL (>= 3)

**Depends** R (>= 3.5)

**Imports** cli, EBImage, graphics, Rdpack, stats, terra

**RdMacros** Rdpack

**Encoding** UTF-8

**Additional\_repositories** <https://bioconductor.org/packages/release/bioc>

**Remotes** bioc::EBImage

**Roxygen** list(markdown = TRUE)

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), vdiff

**Config/testthat/edition** 3

**URL** <https://github.com/pabl1t0x/SatRbedo>,  
<https://pabl1t0x.github.io/SatRbedo/>

**BugReports** <https://github.com/pabl1t0x/SatRbedo/issues>

**VignetteBuilder** knitr

**Config/roxygen2/version** 8.0.0

**Config/pak/sysreqs** cmake libfftw3-dev libgdal-dev gdal-bin libgeos-dev make libjpeg-dev libpng-dev libtiff-dev libuv1-dev libproj-dev libsqlite3-dev

**Repository** <https://pabl1t0x.r-universe.dev>

**Date/Publication** 2026-06-07 15:23:05 UTC

**RemoteUrl** <https://github.com/pabl1t0x/SatRbedo>

**RemoteRef** HEAD

**RemoteSha** b92813fb8d1f38d19f75efead0784b99b74c76a0

## Contents

albedo_broad . . . . .	2
albedo_sat . . . . .	4
cast_shadows . . . . .	7
cfactor_BRDF . . . . .	8
f_BRDF . . . . .	9
hill_shade . . . . .	12
NDSII_hist . . . . .	13
preproc . . . . .	14
shadow_removal . . . . .	15
snow_or_ice . . . . .	16
topo_corr . . . . .	17
topo_splot . . . . .	19
<b>Index</b>	<b>21</b>

---

albedo_broad	<i>Narrow-to-broadband albedo conversion</i>
--------------	--

---

## Description

This function converts narrowband to broadband albedo of snow and ice surfaces using the empirical relationships developed by Knap et al. (1999), Liang (2001), and Feng et al. (2023).

## Usage

```
albedo_Knap(albedo_green, albedo_NIR, saturated = FALSE)
```

```
albedo_Liang(albedo_blue, albedo_red, albedo_NIR, albedo_SWIR1, albedo_SWIR2)
```

```
albedo_Feng(albedo_blue, albedo_green, albedo_red, albedo_NIR)
```

**Arguments**

albedo_green	SpatRaster. Green band albedo (0.53-0.59 um).
albedo_NIR	SpatRaster. Near-infrared band albedo (0.85-0.88 um).
saturated	logical. If TRUE, the green band is saturated, and an expression that is only a function of the near-infrared band is used.
albedo_blue	SpatRaster. Blue band albedo (0.43-0.45 um).
albedo_red	SpatRaster. Red band albedo (0.64-0.67 um).
albedo_SWIR1	SpatRaster. Shortwave-infrared band albedo (1.57-1.65 um).
albedo_SWIR2	SpatRaster. Shortwave-infrared band albedo (2.11-2.29 um).

**Value**

broadband albedo (i.e., the albedo integrated over the entire solar spectrum).

**References**

Feng S, Cook JM, Onuma Y, Naegeli K, Tan W, Anesio AM, Benning LG, Tranter M (2023). "Remote sensing of ice albedo using harmonized Landsat and Sentinel 2 datasets: validation." *International Journal of Remote Sensing*, 1–29. doi:10.1080/01431161.2023.2291000.

Knap WH, Reijmer CH, Oerlemans J (1999). "Narrowband to broadband conversion of Landsat TM glacier albedos." *International Journal of Remote Sensing*, **20**(10), 2091–2110. doi:10.1080/014311699212362.

Liang S (2001). "Narrowband to broadband conversions of land surface albedo I: Algorithms." *Remote Sensing of Environment*, **76**(2), 213–238. doi:10.1016/S00344257(00)002054.

**See Also**

[preproc\(\)](#), [albedo\\_sat\(\)](#)

**Examples**

```
library(terra)
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
blue <- system.file("extdata/athabasca_2020253_B02_S30.tif", package = "SatRbedo")
green <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
red <- system.file("extdata/athabasca_2020253_B04_S30.tif", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
swir1 <- system.file("extdata/athabasca_2020253_B11_S30.tif", package = "SatRbedo")
swir2 <- system.file("extdata/athabasca_2020253_B12_S30.tif", package = "SatRbedo")
blue <- preproc(blue, outline)
green <- preproc(green, outline)
red <- preproc(red, outline)
nir <- preproc(nir, outline)
swir1 <- preproc(swir1, outline)
swir2 <- preproc(swir2, outline)
```

```
# Broadband albedo using Knap et al. (1999)
albedo_Knap(green, nir)

# Broadband albedo using Liang (2001)
albedo_Liang(blue, red, nir, swir1, swir2)

# Broadband albedo using Feng et al. (2023)
albedo_Feng(blue, green, red, nir)
```

---

albedo\_sat

*Satellite albedo retrieval*

---

## Description

This function calculates narrowband and broadband albedo from surface reflectance data. The albedo retrieval method includes corrections for the anisotropic behavior of the reflected radiation field over snow and ice, and narrow-to-broadband albedo conversion algorithms.

## Usage

```
albedo_sat(
  SAA,
  SZA,
  VAA,
  VZA,
  slope,
  aspect,
  method,
  blue = NULL,
  green = NULL,
  red = NULL,
  NIR = NULL,
  SWIR1 = NULL,
  SWIR2 = NULL,
  th
)
```

## Arguments

SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
VAA	SpatRaster or numeric. Satellite sensor azimuth angle in degrees.
VZA	SpatRaster or numeric. Satellite sensor zenith angle in degrees.
slope	SpatRaster or numeric. Surface slope in degrees.
aspect	SpatRaster or numeric. Surface aspect in degrees.

method	character. Number of spectral bands to calculate broadband albedo. There are three options available: "twobands" (green and near-infrared), "fourbands" (blue, green, red, near-infrared), and "fivebands" (blue, red, near-infrared, shortwave-infrared 1, and shortwave-infrared 2). If method="twobands" or method="fivebands", surface reflectance data is corrected for anisotropy before generating broadband albedo. In contrast, method="fourbands" assumes Lambertian reflection and converts surface reflectance to broadband albedo directly. More details and examples about these workflows can be found in the references below.
blue	SpatRaster. Blue band surface reflectance (0.43-0.45 um). Required for method="fourbands" and method="fivebands".
green	SpatRaster. Green band surface reflectance (0.53-0.59 um). Required for all methods.
red	SpatRaster. Red band surface reflectance (0.64-0.67 um). Required for method="fourbands" and method="fivebands".
NIR	SpatRaster. Near-infrared surface reflectance (0.85-0.88 um). Required for all methods.
SWIR1	SpatRaster. Shortwave-infrared surface reflectance (1.57-1.65 um). Required for method="fivebands".
SWIR2	SpatRaster. Shortwave-infrared surface reflectance (2.11-2.29 um). Required for method="fivebands".
th	numeric. NDSII threshold to discriminate between snow and ice.

### Value

Returns a SpatRaster with four layers for method="twobands", five layers for method="fourbands" and six layers for method="fivebands". These layers are:

- If method="twobands": green and NIR narrowband albedo, broadband albedo, and quality flags indicating whether broadband albedo was calculated using the corrected green and NIR narrowband albedos (flag=1) or the NIR albedo only (flag=2). Broadband albedo values higher than one are not excluded for this method.
- If method="fourbands": blue, green, red, and NIR surface reflectance and broadband albedo. Broadband albedo values higher than one and lower than zero are masked out.
- If method="fivebands": blue, red, NIR, SWIR1, SWIR2 narrowband albedo, and broadband albedo. Broadband albedo values higher than one and lower than zero are masked out.

### Notes

If method="fourbands" is used, it is not necessary to call the following arguments: SAA, SZA, VAA, VZA, slope, aspect, and th.

### References

Feng S, Cook JM, Anesio AM, Benning LG, Tranter M (2023). "Long time series (1984–2020) of albedo variations on the Greenland ice sheet from harmonized Landsat and Sentinel 2 imagery." *Journal of Glaciology*, 1–16. doi:10.1017/jog.2023.11.

Klok EJ, Greuell W, Oerlemans J (2003). “Temporal and spatial variation of the surface albedo of Morteratschgletscher, Switzerland, as derived from 12 Landsat images.” *Journal of Glaciology*, **49**(167), 491–502. doi:10.3189/172756503781830395.

Ren S, Miles ES, Jia L, Menenti M, Kneib M, Buri P, McCarthy MJ, Shaw TE, Yang W, Pellicciotti F (2021). “Anisotropy parameterization development and evaluation for glacier surface albedo retrieval from satellite observations.” *Remote Sensing*, **13**(9), 1714. doi:10.3390/rs13091714.

### See Also

[preproc\(\)](#), [snow\\_or\\_ice\(\)](#), [f\\_BRDF\(\)](#), [albedo\\_Knap\(\)](#), [albedo\\_Liang\(\)](#), [albedo\\_Feng\(\)](#), [terra::terrain\(\)](#), [terra::plot\(\)](#)

### Examples

```
library(terra)
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
blue <- system.file("extdata/athabasca_2020253_B02_S30.tif", package = "SatRbedo")
green <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
red <- system.file("extdata/athabasca_2020253_B04_S30.tif", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
swir1 <- system.file("extdata/athabasca_2020253_B11_S30.tif", package = "SatRbedo")
swir2 <- system.file("extdata/athabasca_2020253_B12_S30.tif", package = "SatRbedo")
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
blue <- preproc(grd = blue, outline = outline)
green <- preproc(grd = green, outline = outline)
red <- preproc(grd = red, outline = outline)
nir <- preproc(grd = nir, outline = outline)
swir1 <- preproc(grd = swir1, outline = outline)
swir2 <- preproc(grd = swir2, outline = outline)
dem <- preproc(grd = dem, outline = outline)
SAA <- 164.8
SZA <- 48.9
VAA <- 287.1
VZA <- 7.2
slope <- terra::terrain(dem, v = "slope", neighbors = 4, unit = "degrees")
aspect <- terra::terrain(dem, v = "aspect", neighbors = 4, unit = "degrees")

# Broadband albedo using green and near-infrared surface reflectance data
th <- snow_or_ice(green, nir)$th
alb <- albedo_sat(
  SAA = SAA, SZA = SZA, VAA = VAA, VZA = VZA,
  slope = slope, aspect = aspect, method = "twobands",
  green = green, NIR = nir, th = th
)
plot(alb[[3]]) # Broadband albedo
plot(alb[[4]]) # Flags

# Broadband albedo using blue, green, red and near-infrared surface reflectance data
alb <- albedo_sat(
  method = "fourbands",
  blue = blue, green = green, red = red, NIR = nir
```

```

)
plot(alb[[5]]) # Broadband albedo

# Broadband albedo using blue, red, near-infrared, and shortwave-infrared surface reflectance data
th <- snow_or_ice(green, nir)$th
alb <- albedo_sat(
  SAA = SAA, SZA = SZA, VAA = VAA, VZA = VZA,
  slope = slope, aspect = aspect, method = "fivebands",
  blue = blue, green = green, red = red, NIR = nir, SWIR1 = swir1, SWIR2 = swir2, th = th
)
plot(alb[[6]]) # Broadband albedo

```

---

cast\_shadows

*Cast shadows*


---

### Description

Calculates cast shadows over a matrix or SpatRaster digital elevation model (DEM) for a given illumination direction.

### Usage

```
cast_shadows(dem, SZA, SAA, dl = 0, sombra = dem)
```

### Arguments

dem	matrix or SpatRaster. Digital elevation model representing terrain elevation on a regular grid.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
dl	numeric. Grid spacing. Not needed if dem is SpatRaster.
sombra	Returned matrix or SpatRaster.

### Details

cast\_shadows() calls a fortran routine called doshade that scans the DEM in lines parallel to the sun direction. It compares the projection of the grid cells on a plane perpendicular to the sun to determine whether they are exposed to direct solar illumination or shadowed by neighbor objects, such as mountains. See Corripio (2003) for details. The doshade subroutine was originally written by Javier Corripio and it was shipped with the insol package. This function has been optimized and updated to take advantage of the infrastructure provided by the terra package.

### Value

Returns an object of the same class as the input DEM (matrix or SpatRaster), with values of 0 for cast-shadowed pixels and 1 for not-shaded pixels.

## References

Corripio JG (2003). “Vectorial algebra algorithms for calculating terrain parameters from DEMs and solar radiation modelling in mountainous terrain.” *International Journal of Geographical Information Science*, 17(1), 1–23. doi:10.1080/713811744.

## Examples

```
library(terra)
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
dem <- terra::rast(dem)
SZA <- 49
SAA <- 165
cs <- cast_shadows(dem, SZA, SAA)
plot(cs, col = grey(0:1))
```

---

cfactor\_BRDF

*c-factor BRDF normalization*

---

## Description

Calculates the c-factor proposed by Roy et al. (2016) which can be used to normalize surface reflectance data to nadir BRDF adjusted reflectance (NBAR), or to convert NBAR observations to off-nadir surface reflectance. The c-factor is calculated based on a spectral kernel-based BRDF model. The mathematical formulation of the kernels is presented in Wanner et al. (1995), Lucht et al. (2000) and Carlsen et al. (2020).

## Usage

```
cfactor_BRDF(SAA, SZA, VAA, VZA, NSZ, br = 1, hb = 2, band)
```

## Arguments

SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
VAA	SpatRaster or numeric. Satellite sensor azimuth angle in degrees.
VZA	SpatRaster or numeric. Satellite sensor zenith angle in degrees.
NSZ	numeric. NBAR solar zenith angle. This prescribed NSZ is obtained from the satellite product metadata or calculated using the algorithm and code provided by Li et al. (2019).
br	numeric. Crown relative height parameter. Defaults to 1.0.
hb	numeric. Shape parameter. Defaults to 2.0.
band	character. Spectral band to be processed. There are six bands that can be considered: "blue (0.43-0.45 um)", "green (0.53-0.59 u)", "red (0.64-0.67 um)", "NIR" (near-infrared, 0.85-0.88 um), "SWIR" (shortwave-infrared, 1.57-1.65 um), and "SWIR2 (shortwave-infrared, 2.11-2.29 um)".

**Value**

Returns a single numeric value or a SpatRaster with the c-factor values for a spectral band.

**References**

Carlsen T, Birnbaum G, Ehrlich A, Helm V, Jäkel E, Schäfer M, Wendisch M (2020). “Parameterizing anisotropic reflectance of snow surfaces from airborne digital camera observations in Antarctica.” *The Cryosphere*, **14**(11), 3959–3978. doi:10.5194/tc1439592020.

Li Z, Zhang HK, Roy DP (2019). “Investigation of Sentinel-2 Bidirectional Reflectance Hot-Spot Sensing Conditions.” *IEEE Transactions on Geoscience and Remote Sensing*, **57**(6), 3591–3598. doi:10.1109/TGRS.2018.2885967.

Lucht W, Schaaf CB, Strahler AH (2000). “An algorithm for the retrieval of albedo from space using semiempirical BRDF models.” *IEEE Transactions on Geoscience and Remote Sensing*, **38**(2), 977–998. doi:10.1109/36.841980.

Roy DP, Zhang HK, Ju J, Gomez-Dans JL, Lewis PE, Schaaf CB, Sun Q, Li J, Huang H, Kovalskyy V (2016). “A general method to normalize Landsat reflectance data to nadir BRDF adjusted reflectance.” *Remote Sensing of Environment*, **176**, 255–271. doi:10.1016/j.rse.2016.01.023.

Wanner W, Li X, Strahler AH (1995). “On the derivation of kernels for kernel-driven models of bidirectional reflectance.” *Journal of Geophysical Research*, **100**(D10), 21077. doi:10.1029/95JD02371.

**Examples**

```
# Using data with numeric format
cfactor_BRDF(SAA = 56.58, SZA = 35.6, VAA = 236.4, VZA = 1.00, NSZ = 34.97, band = "NIR")

# Using SpatRaster data
library(terra)
m1 <- matrix(seq(from = 56.54, to = 56.67, length.out = 9), nrow = 3, ncol = 3)
m2 <- matrix(seq(from = 35.54, to = 35.65, length.out = 9), nrow = 3, ncol = 3)
m3 <- matrix(seq(from = 221.05, to = 259.06, length.out = 9), nrow = 3, ncol = 3)
m4 <- matrix(seq(from = 0.53, to = 1.12, length.out = 9), nrow = 3, ncol = 3)
SAA <- terra::rast(m1)
SZA <- terra::rast(m2)
VAA <- terra::rast(m3)
VZA <- terra::rast(m4)
cfactor_BRDF(SAA, SZA, VAA, VZA, NSZ = 34.97, band = "NIR")
```

**Description**

This tool corrects for the anisotropy of reflected radiation by melting snow and ice. For a given surface type, the correction is carried out using an empirical model of the Bidirectional Reflectance Distribution Function (BRDF) that depends on the wavelength bands and the view-solar geometry. The procedure is applicable to visible, near-infrared (NIR), and shortwave-infrared (SWIR) wavelengths. More details about the parameterizations can be found in the references list.

**Usage**

```
f_BRDF(SAA, SZA, VAA, VZA, slope, aspect, method, green, NIR, th)
```

**Arguments**

SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
VAA	SpatRaster or numeric. Satellite sensor azimuth angle in degrees.
VZA	SpatRaster or numeric. Satellite sensor zenith angle in degrees.
slope	SpatRaster or numeric. Surface slope in degrees.
aspect	SpatRaster or numeric. Surface aspect in degrees.
method	character. Number of spectral bands considered for the anisotropy correction. There are two options available: "twobands" (green and NIR) using the parameterizations of Greuell and De Ruyter De Wildt (1999) for ice and Koks (2001) for snow, and "fivebands" (blue, red, NIR, SWIR1, and SWIR2) using the methods described in Ren et al. (2021).
green	SpatRaster. Green band surface reflectance (0.53-0.59 um).
NIR	SpatRaster. Near-infrared band surface reflectance (0.85-0.88 um).
th	numeric. NDSII threshold to discriminate between snow and ice.

**Value**

Returns a SpatRaster with two layers for method="twobands" and five layers for method="fivebands". Each layer contains the anisotropic reflection factors (f) for each spectral band and surface type (snow or ice). All f values represent the difference between the satellite albedo retrievals and the surface reflectance observations.

**Notes**

f is not calculated for method="twobands" if the green or the NIR surface reflectance values are greater than one or for pixels for which the SZA relative to the inclined surface is  $>65.51^\circ$  over snow or  $>74.2^\circ$  over ice. For method="fivebands", f is not calculated if the SZA relative to the inclined surface is  $>70.9^\circ$  over snow or  $>57.6^\circ$  over ice.

## References

De Ruyter De Wildt MS, Oerlemans J, Björnsson H (2002). “A method for monitoring glacier mass balance using satellite albedo measurements: application to Vatnajökull, Iceland.” *Journal of Glaciology*, **48**(161), 267–278. doi:10.3189/172756502781831458.

Greuell W, De Ruyter De Wildt M (1999). “Anisotropic reflection by melting glacier ice: measurements and parametrizations in Landsat TM Bands 2 and 4.” *Remote Sensing of Environment*, **70**(3), 265–277. doi:10.1016/S00344257(99)000437.

Koks M (2001). *Anisotropic reflection of radiation by melting snow. Landsat TM bands 2 and 4*. Master’s thesis, Universiteit Utrecht. Instituut voor Marien en Atmosferisch Onderzoek Utrecht (IMAU).

Ren S, Miles ES, Jia L, Menenti M, Kneib M, Buri P, McCarthy MJ, Shaw TE, Yang W, Pellicciotti F (2021). “Anisotropy parameterization development and evaluation for glacier surface albedo retrieval from satellite observations.” *Remote Sensing*, **13**(9), 1714. doi:10.3390/rs13091714.

## See Also

[preproc\(\)](#), [terra::terrain\(\)](#), [snow\\_or\\_ice\(\)](#), [terra::plot\(\)](#)

## Examples

```
library(terra)
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
green <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
green <- preproc(grd = green, outline = outline)
nir <- preproc(grd = nir, outline = outline)
dem <- preproc(grd = dem, outline = outline)
SAA <- 164.8
SZA <- 48.9
VAA <- 287.1
VZA <- 7.2
slope <- terra::terrain(dem, v = "slope", neighbors = 4, unit = "degrees")
aspect <- terra::terrain(dem, v = "aspect", neighbors = 4, unit = "degrees")

# f coefficients for two bands
th <- snow_or_ice(green, nir)$th
f <- f_BRDF(
  SAA = SAA, SZA = SZA, VAA = VAA, VZA = VZA,
  slope = slope, aspect = aspect,
  method = "twobands", green = green, NIR = nir,
  th = th
)
plot(f[[1]])
plot(f[[2]])

# f coefficients for five bands
th <- snow_or_ice(green, nir)$th
```

```
f <- f_BRDF(  
  SAA = SAA, SZA = SZA, VAA = VAA, VZA = VZA,  
  slope = slope, aspect = aspect,  
  method = "fivebands", green = green, NIR = nir,  
  th = th  
)  
plot(f[[1]])  
plot(f[[2]])  
plot(f[[3]])  
plot(f[[4]])  
plot(f[[5]])
```

---

hill\_shade

*Topographic hillshading*

---

### Description

Computes the hypothetical illumination of a surface by setting the position of the sun and calculating the illumination values for each grid cell of a DEM. The `hill_shade` function is based upon the vectorial algebra algorithms developed by Corripio (2003). This function has been optimized and updated to take advantage of the infrastructure provided by the `terra` package.

### Usage

```
hill_shade(dem, SZA, SAA)
```

### Arguments

dem	SpatRaster. Digital elevation model representing terrain elevation on a regular grid.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
SAA	SpatRaster or numeric. Solar azimuth angle in degrees.

### Value

Returns a hillshade map in SpatRaster format.

### References

Corripio JG (2003). "Vectorial algebra algorithms for calculating terrain parameters from DEMs and solar radiation modelling in mountainous terrain." *International Journal of Geographical Information Science*, **17**(1), 1–23. doi:10.1080/713811744.

### See Also

[normal\\_vector\(\)](#), [sun\\_vector\(\)](#)

**Examples**

```
library(terra)
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
dem <- terra::rast(dem)
SZA <- 49
SAA <- 165
hs <- hill_shade(dem, SZA, SAA)
plot(hs, col = grey(1:100 / 100))
```

---

NDSII\_hist

*NDSII histogram*


---

**Description**

This function creates an NDSII histogram with a vertical line showing the selected threshold discriminating snow and ice.

**Usage**

```
NDSII_hist(NDSII, th, breaks = 64, stdev = 2)
```

**Arguments**

NDSII	SpatRaster. Normalized Difference Snow Ice Index (NDSII).
th	numeric. NDSII threshold to discriminate between snow and ice.
breaks	one of: <ul style="list-style-type: none"> <li>• a vector giving the breakpoints between histogram cells,</li> <li>• a function to compute the vector of breakpoints,</li> <li>• a single number giving the number of cells for the histogram,</li> <li>• a character string naming an algorithm to compute the number of cells (see ‘Details’),</li> <li>• a function to compute the number of cells.</li> </ul> <p>In the last three cases the number is a suggestion only; as the breakpoints will be set to <a href="#">pretty</a> values, the number is limited to 1e6 (with a warning if it was larger). If breaks is a function, the x vector is supplied to it as the only argument (and the number of breaks is only limited by the amount of available memory).</p>
stdev	numeric. Standard deviation cutoff value for histogram stretching.

**See Also**

[preproc\(\)](#), [snow\\_or\\_ice\(\)](#), [graphics::hist\(\)](#)

## Examples

```
green <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
green <- preproc(grd = green, outline = outline)
nir <- preproc(grd = nir, outline = outline)
res <- snow_or_ice(green, nir)
NDSII_hist(res$NDSII, res$th, breaks = 16, stdev = 3)
```

---

```
preproc
```

```
Satellite data pre-processing
```

---

## Description

preproc() crops an input grid to a specified extent with a polygon or SpatExtent, reprojects the grid to a specified coordinate system, and converts the data from integer to floating point using an offset and a scale factor.

## Usage

```
preproc(
  grd,
  drivers = NULL,
  outline = NULL,
  coords = NULL,
  add_offset = 0,
  scale_factor = 1
)
```

## Arguments

grd	filename (character) or SpatRaster of the raster to be processed, in one of GDAL's raster <b>drivers</b> .
drivers	character. GDAL drivers to consider
outline	filename (character) of the shapefile containing the outline of the area of interest, in .shp format, or SpatExtent giving a vector of length four (xmin, xmax, ymin, ymax).
coords	filename (character) or SpatRaster of the raster with the coordinate system definition. Alternatively, a coordinate reference system (CRS) description can be provided. In this case, the CRS can be described using the PROJ-string (e.g., "+proj=longlat +datum=WGS84") or the WKT format (e.g., "EPSG:4326").
add_offset	band-specific offset added to each grid value. Defaults to zero.
scale_factor	band-specific scale factor applied to each grid value. Defaults to one.

**Value**

Returns a pre-processed SpatRaster grid.

**See Also**

[terra::rast\(\)](#), [terra::ext\(\)](#), and [terra::project\(\)](#) which this function wraps.

**Examples**

```
# uncorrected grid
f <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
preproc(grd = f)

# crop grid
g <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
v <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
preproc(grd = g, outline = v)

# crop and reproject grid
g <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
v <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
preproc(grd = g, outline = v, coords = "+proj=longlat +datum=WGS84")

preproc(grd = g, outline = v, coords = "EPSG:4326")

# Transform grid values
library(terra)
g <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
orig <- preproc(grd = g)
minmax(orig)
new <- preproc(grd = g, add_offset = 5, scale_factor = 10)
minmax(new)
```

---

shadow\_removal

*Topographic shadows*

---

**Description**

Detects and removes self-shadowed pixels (i.e., pixels having aspect angles oriented away from the sun) and cast shadows (i.e, pixels facing towards the direction of the sun but where sunlight is blocked by neighbor topographic features such as mountains). Self-shadowed pixels are detected when the angle between the sun and the vector normal to the grid cell's surface is higher than  $\frac{\pi}{2}$ . Cast shadows are calculated using the [cast\\_shadows\(\)](#) function.

**Usage**

```
shadow_removal(dem, SZA, SAA, mask = TRUE)
```

**Arguments**

dem	SpatRaster. Digital elevation model representing terrain elevation on a regular grid.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
mask	logical. If TRUE, shadow pixels are masked.

**Value**

Returns a SpatRaster with topographic shadows removed.

**See Also**

[cast\\_shadows\(\)](#), [hill\\_shade\(\)](#)

**Examples**

```
library(terra)
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
dem <- terra::rast(dem)
SZA <- 49
SAA <- 165
msk <- shadow_removal(dem, SZA, SAA, mask = TRUE)
plot(msk, col = "blue")

# Overlay an RGB composite with the shadow masks
blue <- preproc(system.file("extdata/athabasca_2020253_B02_S30.tif", package = "SatRbedo"))
green <- preproc(system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo"))
red <- preproc(system.file("extdata/athabasca_2020253_B04_S30.tif", package = "SatRbedo"))
r <- c(stretch(blue), stretch(green), stretch(red))
RGB(r) <- c(3, 2, 1)
plot(r)
msk <- shadow_removal(dem, SZA, SAA, mask = FALSE)
plot(msk, col = grey(0:1), alpha = 0.2, add = TRUE)
```

---

snow\_or\_ice

*Automatic discrimination of snow and ice cover types*

---

**Description**

Calculates the Normalized Difference Snow Ice Index (NDSII) (Keshri et al. 2009) to discriminate between snow and ice surfaces. The discrimination is performed using an automatic threshold selection method based on the Otsu algorithm (Otsu 1979).

**Usage**

```
snow_or_ice(green, NIR)
```

**Arguments**

green           SpatRaster. Green band surface reflectance (0.53-0.59  $\mu\text{m}$ ).  
 NIR             SpatRaster. Near-infrared band surface reflectance (0.85-0.88  $\mu\text{m}$ ).

**Value**

Returns a list with two objects. The first component is an NDSII SpatRaster and the second component provides the optimal threshold to discriminate between snow and ice surfaces.

**References**

Keshri AK, Shukla A, Gupta RP (2009). "ASTER ratio indices for supraglacial terrain mapping." *International Journal of Remote Sensing*, **30**(2), 519–524. doi:10.1080/01431160802385459.

Otsu N (1979). "A Threshold Selection Method from Gray-Level Histograms." *IEEE Transactions on Systems, Man, and Cybernetics*, **9**(1), 62–66. doi:10.1109/TSMC.1979.4310076.

**See Also**

[preproc\(\)](#), [EBImage::otsu\(\)](#), [terra::plot\(\)](#)

**Examples**

```
green <- system.file("extdata/athabasca_2020253_B03_S30.tif", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
green <- preproc(grd = green, outline = outline)
nir <- preproc(grd = nir, outline = outline)
res <- snow_or_ice(green, nir)

# Plot NDSII
library(terra)
plot(res$NDSII, range = c(-0.5, 0.5))

# Plot a cover type binary map (snow/ice)
library(terra)
plot(res$NDSII > res$th, type = "classes", col = c("#FFFC8", "#00407F"), levels = c("snow", "ice"))
```

---

topo\_corr

*Topographic correction*


---

**Description**

This tool corrects surface reflectance for the effects of topography. The objective of the topographic correction is to compensate for the differences in solar illumination to minimize the variability of observed reflectance for similar targets and obtain the equivalent reflectance values over flat terrain. The empirical methods provided here are suitable for mountain environments with rugged topography and non-Lambertian surface properties. More details about these methods can be found in the references below.

**Usage**

```
topo_corr(band, dem, SAA, SZA, method = "tanrotation", IC_min = NULL)
```

**Arguments**

band	SpatRaster. Spectral band to be corrected.
dem	SpatRaster. Digital elevation model representing terrain elevation on a regular grid.
SAA	SpatRaster or numeric. Solar azimuth angle in degrees.
SZA	SpatRaster or numeric. Solar zenith angle in degrees.
method	character. Topographic correction model to be used. Two options are available: "tanrotation" (implements the rotation model proposed by Tan et al. 2010), and "ccorrection" (C model, Teillet et al. 1982).
IC_min	numeric. Minimum incidence angle to mask out regions of low illumination.

**Value**

Returns a list with two objects. The first component is a SpatRaster with two layers providing the illumination condition (IC) and the corrected surface reflectance, and the second component contains the values of the empirical parameters obtained by each topographic correction model.

**References**

- Chen R, Yin G, Zhao W, Yan K, Wu S, Hao D, Liu G (2023). "Topographic Correction of Optical Remote Sensing Images in Mountainous Areas: A systematic review." *IEEE Geoscience and Remote Sensing Magazine*, **11**(4), 125–145. doi:10.1109/MGRS.2023.3311100.
- Riaño D, Chuvieco E, Salas J, Aguado I (2003). "Assessment of different topographic corrections in Landsat-TM data for mapping vegetation types." *IEEE Transactions on Geoscience and Remote Sensing*, **41**(5), 1056–1061. doi:10.1109/TGRS.2003.811693.
- Tan B, Masek JG, Wolfe R, Gao F, Huang C, Vermote EF, Sexton JO, Ederer G (2013). "Improved forest change detection with terrain illumination corrected Landsat images." *Remote Sensing of Environment*, **136**, 469–483. doi:10.1016/j.rse.2013.05.013.
- Tan B, Wolfe R, Masek J, Gao F, Vermote EF (2010). "An illumination correction algorithm on Landsat-TM data." In *2010 IEEE International Geoscience and Remote Sensing Symposium*, 1964–1967. doi:10.1109/IGARSS.2010.5653492.
- Teillet PM, Guindon B, Goodenough DG (1982). "On the Slope-Aspect Correction of Multispectral Scanner Data." *Canadian Journal of Remote Sensing*, **8**(2), 84–106. doi:10.1080/07038992.1982.10855028.

**See Also**

[preproc\(\)](#), [landsat::topocorr\(\)](#)

## Examples

```
library(terra)
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
nir <- preproc(grd = nir)
dem <- preproc(grd = dem)
SAA <- 164.8
SZA <- 48.9

# Topographic correction using method="tanrotation"
corr <- topo_corr(band = nir, dem = dem, SAA = SAA, SZA = SZA, method = "tanrotation")
plot(corr$bands[[1]]) # plot IC
plot(corr$bands[[2]]) # plot corrected surface reflectance

# Topographic correction using method="ccorrection"
corr <- topo_corr(band = nir, dem = dem, SAA = SAA, SZA = SZA, method = "ccorrection", IC_min = 0.3)
plot(corr$bands[[2]]) # plot corrected surface reflectance
```

---

topo\_splot

*Reflectance-IC scatterplot*

---

## Description

This function creates a scatterplot of surface reflectance vs. illumination condition (IC). The plot can be used to explore the dependency of the original and the topographically-corrected surface reflectance data on IC.

## Usage

```
topo_splot(IC, band, maxcell = 1e+05)
```

## Arguments

IC	SpatRaster. Illumination condition.
band	SpatRaster. Spectral band to be processed.
maxcell	positive integer. Maximum number of cells to use for the plot

## See Also

[topo\\_corr\(\)](#), [graphics::abline\(\)](#), [stats::lm\(\)](#)

**Examples**

```
outline <- system.file("extdata/athabasca_outline.shp", package = "SatRbedo")
nir <- system.file("extdata/athabasca_2020253_B8A_S30.tif", package = "SatRbedo")
dem <- system.file("extdata/athabasca_dem.tif", package = "SatRbedo")
nir <- preproc(grd = nir)
dem <- preproc(grd = dem)
SAA <- 164.8
SZA <- 48.9
corr <- topo_corr(band = nir, dem = dem, SAA = SAA, SZA = SZA, method = "tanrotation")

# Scatterplot of IC vs. uncorrected surface reflectance
topo_splot(corr$bands[[1]], nir)

# Scatterplot of IC vs. topographically-corrected surface reflectance
topo_splot(corr$bands[[1]], corr$bands[[2]])
```

# Index

albedo\_broad, [2](#)  
albedo\_Feng (albedo\_broad), [2](#)  
albedo\_Feng(), [6](#)  
albedo\_Knap (albedo\_broad), [2](#)  
albedo\_Knap(), [6](#)  
albedo\_Liang (albedo\_broad), [2](#)  
albedo\_Liang(), [6](#)  
albedo\_sat, [4](#)  
albedo\_sat(), [3](#)

cast\_shadows, [7](#)  
cast\_shadows(), [15](#), [16](#)  
cfactor\_BRDF, [8](#)

EImage::otsu(), [17](#)

f\_BRDF, [9](#)  
f\_BRDF(), [6](#)

graphics::abline(), [19](#)  
graphics::hist(), [13](#)

hill\_shade, [12](#)  
hill\_shade(), [16](#)

landsat::topocorr(), [18](#)

NDSII\_hist, [13](#)  
normal\_vector(), [12](#)

preproc, [14](#)  
preproc(), [3](#), [6](#), [11](#), [13](#), [17](#), [18](#)  
pretty, [13](#)

shadow\_removal, [15](#)  
snow\_or\_ice, [16](#)  
snow\_or\_ice(), [6](#), [11](#), [13](#)  
stats::lm(), [19](#)  
sun\_vector(), [12](#)

terra::ext(), [15](#)  
terra::plot(), [6](#), [11](#), [17](#)  
terra::project(), [15](#)  
terra::rast(), [15](#)  
terra::terrain(), [6](#), [11](#)  
topo\_corr, [17](#)  
topo\_corr(), [19](#)  
topo\_splot, [19](#)