

The Snow Pilot data service prototype

Version 1.0

<i>Version</i>	<i>Date</i>	<i>Status</i>	<i>Authors</i>
<i>1.0</i>	<i>9/12/2023</i>	<i>First draft</i>	<i>Roberto Salzano, Rosamaria Salvatori</i>

Table of contents

1.	The Data chain components.....	3
1.1	The image pre-screening	3
1.2	The image radial distortion correction	3
1.3	The image orthoprojection.....	4
1.4	The Image segmentation	4
1.5	The Image classification.....	5
1.6	The data product requirements:	6
2	References	7
3	Software packages.....	8

1. The Data chain components

The first key component was concerned on defining the architecture of the data chain and starting processing archived imagery available in the Ny-Ålesund area. The selected datasets include the cameras operating at the Zeppelin Observatory [Pedersen, 2013], maintained by NPI, and the device supported by CNR at the Amundsen-Nobile Climate Change Tower [Salzano et al., 2022]. The structure of the data chain considers different components (Figure 1).

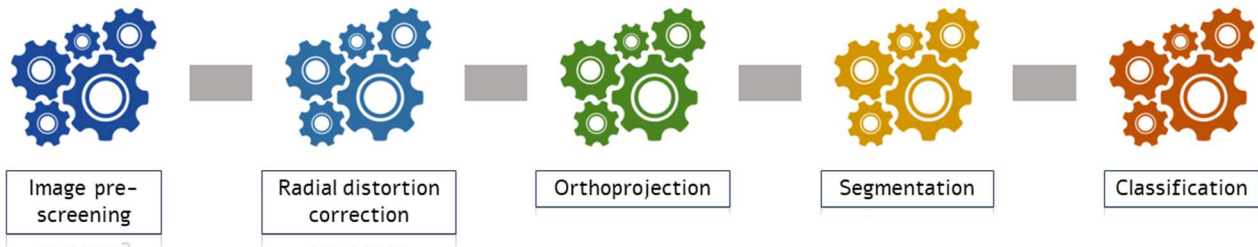


Figure 1 Architecture of the data processing chain

1.1 The image pre-screening

The pre-processing performed on the available imagery is focused on checking metadata associated with the image file, looking at the image size and the pixel resolution. Such a preliminary survey can identify failure in the data corruption and hopefully some issue in the server accessibility. The second step is the analysis of RGB values and their image patterns with the major goal to identify images characterized by low illumination, as well as situation with intense cloud cover. Finally, additional lens interferences (rain drops or ice crusts) are discarded. The procedure was based on the ImageMagick software [ImageMagick Studio LLC., 2023] and on the interface “magick” [Ooms, 2023] used in the R programming environment [R Core Team, 2023].

1.2 The image radial distortion correction

The monoploting of the image on the surface require two key information about the camera orientation and the sensor specifications. The preliminary process consists of calibrating the time-lapse camera using a reference chessboard. This operation is based on having at least 6 images shot by the camera system with the reference chessboard in a close-range distance. Having different chessboard orientations in front of the camera is possible, using standard computer vision techniques [Zhang, 2000], to obtain the intrinsic lens features useful for correcting the lens distortion (Figure 2). We used in this case the MATLAB and the Computer Vision toolbox [The Mathworks, 2023].

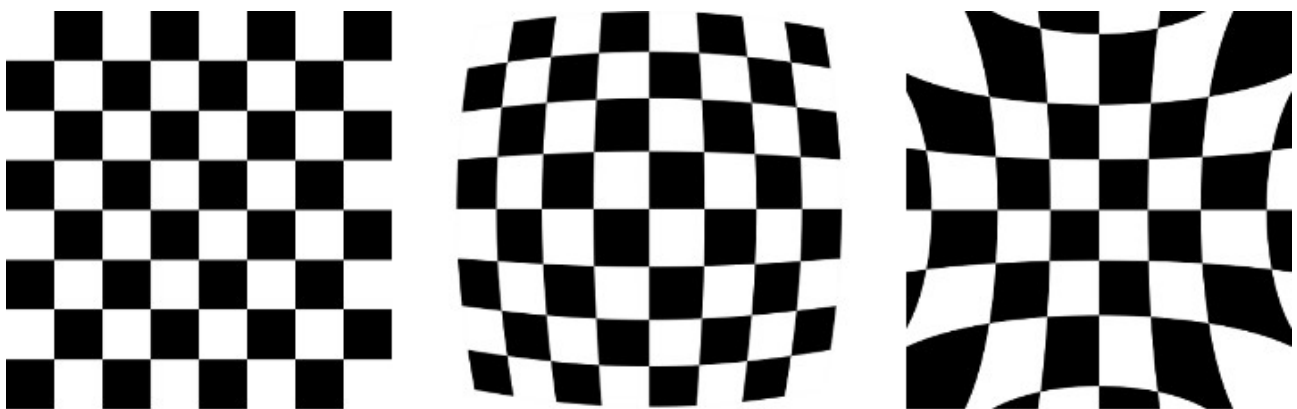


Figure 2. Comparison between undistorted (left), and positively or negatively distorted images.

1.3 The image orthorectification

The orthorectification module is based on the geometrical correction of the perspective view (Figure 3). This step is implemented following [Corripio, 2004]. The available digital elevation model [Porter et al., 2022], with a 5 m spatial resolution and 1 m vertical resolution, provided several topographic points that were projected on the camera view. The effectiveness of the correction was estimated considering the available ground control points. The code was developed in the R programming environment [R Core Team, 2023] using the libraries “terra” [Hijmans, 2023] and “jpeg” [Urbanek, 2022].

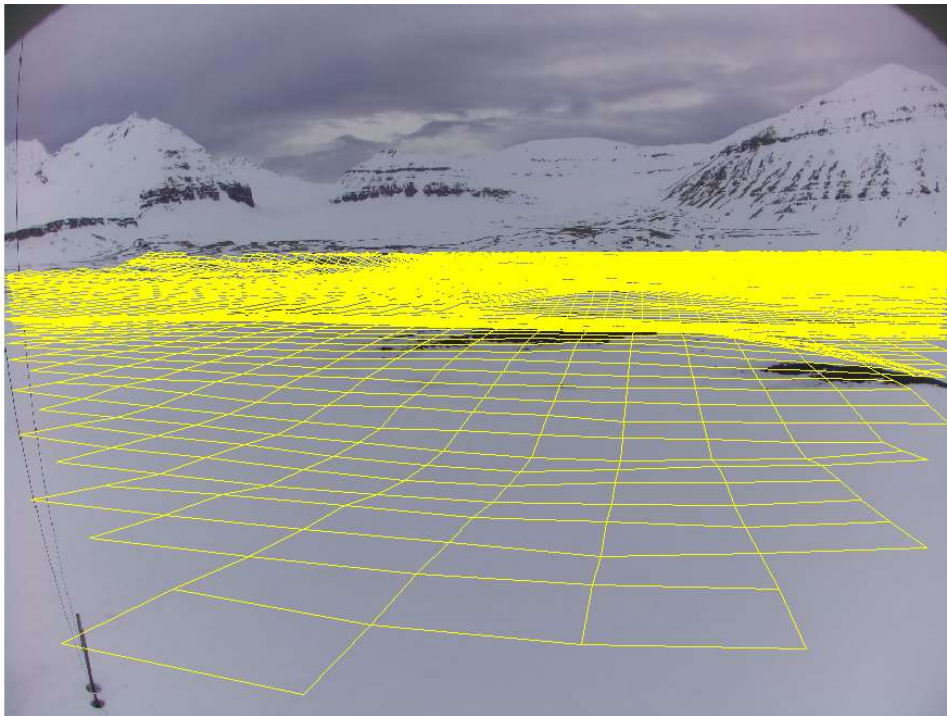


Figure 3. Example of 10m grid projection on the image perspective.

1.4 The Image segmentation

The Segmentation procedure is based on the Spectral Similarity approach, proposed by [Salzano et al., 2019]. The module is based on measuring the spectral variations in a 3D colour space where reference endmembers are a theoretical “white” snow and a theoretical “black” target. The parameters estimated in this vector system are the spectral angle defined by [Kruse et al., 1993] and the Euclidean distance [Jensen, 2015], respectively calculated considering white and black references. While the parameter based on the Spectral Similarity represents an independent spectral feature, the Euclidean distance of the vector can be defined as a brightness-dependent feature. The involvement of all the three-color components will support the increase of surface types that can be discriminated: snow, shadowed snow and not snow. The proposed approach was developed in the R programming environment [R Core Team, 2023] using the libraries “terra” [Hijmans, 2023] and “jpeg” [Urbanek, 2022]. The first step consists of rearranging the three-color components of each pixel into a new two-dimensional vector space, mathematically defined as follows: the spectral angle which represents the relative proportion of the three-pixel components in relationship to the reference composition. The angle varies from zero, which can be associated with a “flat” behaviour of colours ($R = G = B$), to $\pi/2$, referring to a very dissimilar behaviour from the theoretical “white” reference. The spectral distance is conversely an estimation of the vector length in the RGB space. It can range from 0 (black) to 1.73 (white) and it can be associated with the Euclidean distance from a “black” reference RGB composition. While this parameter is sensitive to the brightness of colours, the spectral angle is invariant with brightness [Van der Meer, 2006]. The outcome of this step consists in the frequency counting of pixels considering the two spectral components with a 0.05 resolution (Figure 4). Furthermore, the total number of included pixels and

the area included in the cluster perimeter can be estimated. The second step of the procedure consists in discriminating clusters from the obtained frequency distribution, and a watershed algorithm [Vincent et al., 1991] can support this segmentation phase. Each cluster was fitted with a normal distribution to retrieve modes and deviations. If clusters are very close to each other, they can be combined in one larger group depending on their probability to be discriminated using the Mahalanobis distance. The criteria adopted for the definition of the cluster perimeter was based on the pixel frequency higher than the Poisson error of the adjacent pixel. The procedure for the delimitation of the cluster perimeter was implemented using a per-pixel method following [Seo et al., 2016]. The code was developed in the R programming environment.

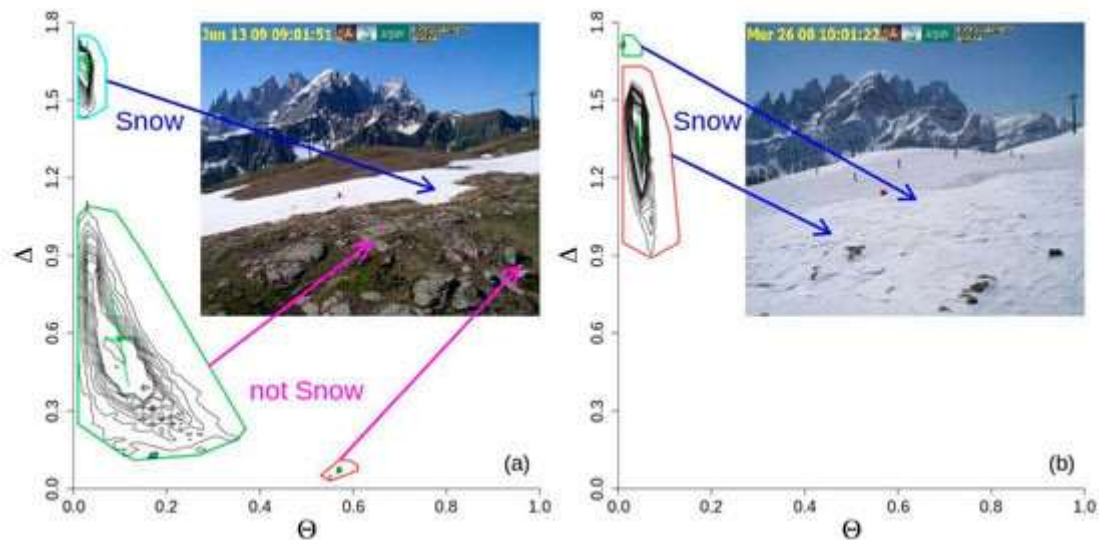


Figure 4. Examples of two different snow-not-snow mixtures [Salzano et al., 2019]. Coloured polygons identified areas of clusters in presence of two different situations: partial (a) and full (b) snow cover. Lower plots are frequency distribution of pixels at the different spectral angles (ϑ) and spectral distances (Δ).

1.5 The Image classification

The final step consists in the identification of the surface type (snow, not snow and shadowed snow). This step was defined observing the frequency distributions of pixels in the defined spectral space. It was possible to detect that snow covers were generally characterized by higher spectral angles and lower distance values than not-snow covers (Figure 5). Snowed centroids (defined by $\mu\Delta$ and $\mu\theta$) were generally positioned where angles were higher than 0.9 and distances were lower than 0.1.

Furthermore, the range of cluster values (Δ_{max} , Δ_{min} , θ_{max} and θ_{min}) are characterized by short distance variations compared to angles in the case of snow-covered surfaces. From this point of view, clusters with limited perimeters and a high number of included pixels describe surfaces with homogeneous reflective behaviour, as expected for snow-covered surfaces. The second rule that can be considered includes clusters with limited perimeters and consistent number of included pixels. The optical behaviour of those clusters must be coupled to their centroid position that must have low spectral angles. These constraints describe, also in this case, clusters characterized by a homogeneous spectral behaviour coherent with a snow-covered surface. The third rule that completes the classification procedure consisted of estimating the range of Δ between the defined clusters in the image and on defining a threshold ($T\Delta$) that discriminates snow and other surface types. Two situations can occur for defining clusters above the threshold as snow-covered surface: one with multiple clusters and one with a single polygon.

The identification of the snow clusters has been checked considering also another automated method, which is a linear classifier based on thresholding of the blue channel (BT), introduced by [Salvatori et al., 2011] in the Snow-noSnow software. The method is based on the frequency counting of the blue component, and its hardness is associated with the definition of snow-not-snow limit looking at increments in the blue-channel histogram. This method has been used in several studies and it has shown some limitations. The illuminating

conditions, the surface roughness and the distance from the camera are critical issues that affect the performance on retrieving snowed covers. The code was developed in the R programming environment using the libraries “terra” [Hijmans, 2023] and “jpeg” [Urbanek, 2022].

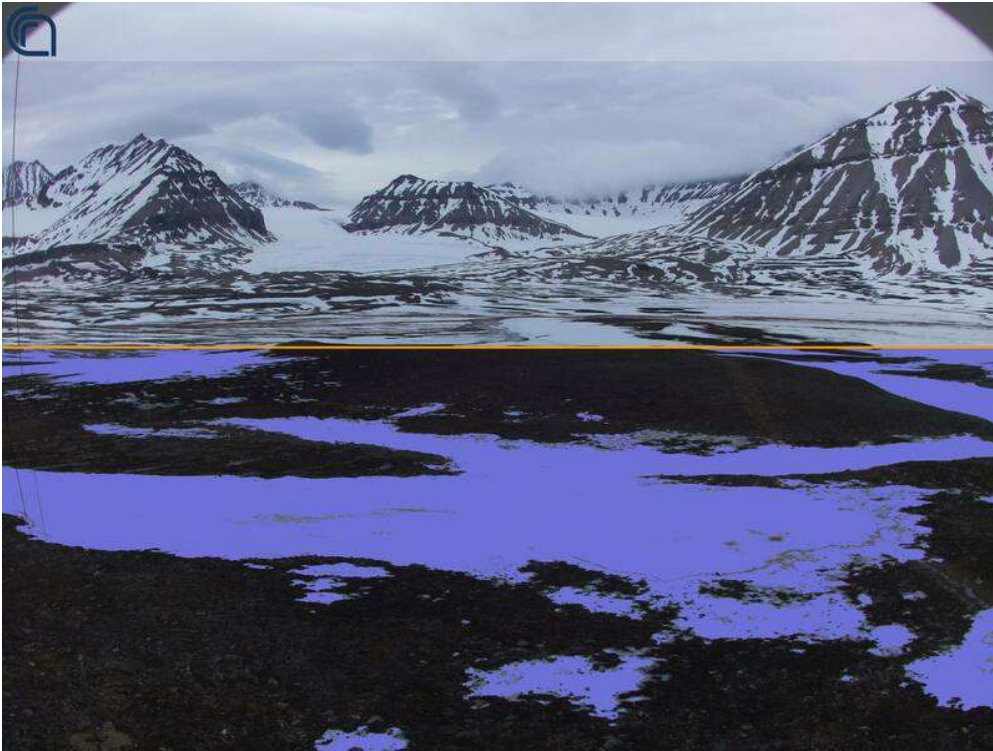


Figure 5. Example of classified image.

1.6 The data product requirements

Combining the classification output with the ortho-projection procedure it is possible to prepare a product concerning the fractional snow-covered area (FSCA). The grid extraction can be based on specific satellite products (Sentinel-2). The data format is the netCDF standard format, where metadata is prepared following the ISO 19115 guidelines and the Climate & Forecast convention.

2 References

- Corripio, J.G. Snow surface albedo estimation using terrestrial photography. *Int. J. Remote Sens.* **2004**, *25*, 5705–5729. <https://doi.org/10.1080/01431160410001709002>
- Porter, C.; Morin, P.; Howat, I.; Noh, M.J.; Bates, B.; Peterman, K.; Keeseey, S.; Schlenk, M.; Gardiner, J.; Tomko, K. ArcticDEM. Harvard Dataverse, **2022**. Available online: <https://www.pgc.umn.edu/data/arcticdem/> (accessed on 22 November 2022).
- Jensen, J.R. *Introductory Digital Image Processing: A Remote Sensing Perspective*, 4th ed. Pearson Series in Geographic Information Science; Pearson: Glenview, IL, USA, **2015**; pp. 1–544.
- Kruse, F.A.; Lefkoff, A.B.; Boardman, J.W.; Heidebrecht, K.B.; Shapiro, A.T.; Barloon, P.J.; Goetz, A.F.H. The spectral image processing system (SIPS)-Interactive visualization and analysis of imaging spectrometer data. *Remote Sens. Environ.* **1993**, *44*, 145–163. [https://doi.org/10.1016/0034-4257\(93\)90013-N](https://doi.org/10.1016/0034-4257(93)90013-N)
- Pedersen, C. Zeppelin web camera time-series [Dataset]. Norwegian Polar Institute, **2013**. <https://doi.org/10.21334/npolar.2013.9fd6dae0>
- Salvatori, R.; Plini, P.; Giusto, M.; Valt, M.; Salzano, R.; Montagnoli, M.; Cagnati, A.; Crepaz, G.; Sigismondi, D. Snow cover monitoring with images from digital camera systems. *Ital. J. Remote Sens.* **2011**, *43*, 137–145.
- Salzano, R.; Salvatori, R.; Valt, M.; Giuliani, G.; Chatenoux, B.; Ioppi, L. Automated Classification of Terrestrial Images: The Contribution to the Remote Sensing of Snow Cover. *Geosciences* **2019**, *9*, 97. <https://doi.org/10.3390/geosciences9020097>
- Salzano, R.; Salvatori, R.; Cerrato, R.; Viola, A.P. Fractional Snow-Covered Area at Ny-Ålesund (Svalbard, Norway), **2022**. <https://doi.org/10.5281/zenodo.5705593>.
- Seo, J.; Chae, S.; Shim, J.; Kim, D.; Cheong, C.; Han, T. Fast Contour-Tracing Algorithm Based on a Pixel-Following Method for Image Sensors. *Sensors* **2016**, *16*, 353. <https://doi.org/10.3390/s16030353>
- Van der Meer, F. The effectiveness of spectral similarity measures for the analysis of hyperspectral imagery. *Int. J. Appl. Earth Obs. Geoinf.* **2006**, *8*, 3–17. <https://doi.org/10.1016/j.jag.2005.06.001>
- Vincent, L.; Soille, P. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 583–598. <https://doi.org/10.1109/34.87344>
- Zhang, Z. "A Flexible New Technique for Camera Calibration." *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22* (11), 1330–34. <https://doi.org/10.1109/34.888718>

3 Software packages

Hijmans, R. “terra: Spatial Data Analysis”. R package version 1.7-29, **2023**. <https://CRAN.R-project.org/package=terra> (accessed on 1 October 2023).

ImageMagick Studio LLC. “ImageMagick”; ver. 7.1.0, **2023**. <https://imagemagick.org> (accessed on 1 October 2023).

Ooms, J. “magick: Advanced Graphics and Image-Processing in R”. R package version 2.7.4, **2023**. <https://CRAN.R-project.org/package=magick> (accessed on 1 October 2023).

The MathWorks Inc. MATLAB version R2022a, Natick, Massachusetts, **2022**. <https://www.mathworks.com> (accessed on 1 October 2023).

R Core Team. R: A Language and Environment for Statistical Computing; ver. 4.3.1; R Foundation for Statistical Computing: Vienna, Austria, **2023**. <https://www.R-project.org/> (accessed on 1 July 2023).

Urbanek S “jpeg: Read and write JPEG images”. R package version 0.1-10, **2022**. <https://CRAN.R-project.org/package=jpeg> (accessed on 1 October 2023).