



Technical Note

# Sentinel-1 SAR Backscatter Analysis Ready Data Preparation in Google Earth Engine

Adugna Mullissa <sup>1,\*</sup>, Andreas Vollrath <sup>2</sup>, Christelle Odongo-Braun <sup>1</sup>, Bart Slagter <sup>1</sup>, Johannes Balling <sup>1</sup>, Yaqing Gou <sup>1</sup>, Noel Gorelick <sup>3</sup> and Johannes Reiche <sup>1</sup>

<sup>1</sup> Laboratory of Geo-Information Science and Remote Sensing, Wageningen University and Research, Droevedaalsesteeg 3, 6708 PB Wageningen, The Netherlands; brauncch@gmail.com (C.O.-B.); bart.slagter@wur.nl (B.S.); johannes.balling@wur.nl (J.B.); yaqing.gou@wur.nl (Y.G.); johannes.reiche@wur.nl (J.R.)

<sup>2</sup> Food and Agriculture Organization of the United Nations, Viale delle Terme di Caracalla, 00153 Roma, Italy; andreas.vollrath@fao.org

<sup>3</sup> Google Inc., Google Switzerland, 8002 Zurich, Switzerland; gorelick@google.com

\* Correspondence: adugna.mullissa@wur.nl

**Abstract:** Sentinel-1 satellites provide temporally dense and high spatial resolution synthetic aperture radar (SAR) imagery. The open data policy and global coverage of Sentinel-1 make it a valuable data source for a wide range of SAR-based applications. In this regard, the Google Earth Engine is a key platform for large area analysis with preprocessed Sentinel-1 backscatter images available within a few days after acquisition. To preserve the information content and user freedom, some preprocessing steps (e.g., speckle filtering) are not applied on the ingested Sentinel-1 imagery as they can vary by application. In this technical note, we present a framework for preparing Sentinel-1 SAR backscatter Analysis-Ready-Data in the Google Earth Engine that combines existing and new Google Earth Engine implementations for additional border noise correction, speckle filtering and radiometric terrain normalization. The proposed framework can be used to generate Sentinel-1 Analysis-Ready-Data suitable for a wide range of land and inland water applications. The Analysis Ready Data preparation framework is implemented in the Google Earth Engine JavaScript and Python APIs.

**Keywords:** Sentinel-1; analysis ready data; Google earth engine; preprocessing; speckle filter



**Citation:** Mullissa, A.; Vollrath, A.; Odongo-Braun, C.; Slagter, B.; Balling, J.; Gou, Y.; Gorelick, N.; Reiche, J. Sentinel-1 SAR Backscatter Analysis-Ready-Data Preparation in Google Earth Engine. *Remote Sens.* **2021**, *13*, 1954. <https://doi.org/10.3390/rs13101954>

Academic Editor: Sang-Eun Park

Received: 23 April 2021

Accepted: 14 May 2021

Published: 17 May 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Global coverage satellite data are considered indispensable for earth monitoring due to their large scale availability and low cost. In this regard, the Sentinel-1 synthetic aperture radar (SAR) data from the European Union's Copernicus program is unique, as it, for the first time, provides temporally dense and high spatial resolution satellite SAR imagery freely for the entire globe.

Sentinel-1's cloud-penetrating radar provides gap-free observations in almost all weather, day and night and is considered a valuable data source for monitoring the earth's surface particularly in regions with year round cloud cover. Examples of large-scale Sentinel-1 SAR applications include forest disturbance monitoring [1], rice cropland mapping [2], ground deformation monitoring [3], and inland water body monitoring [4].

In the 2010s, Google introduced Google Earth Engine (GEE) to further enhance the usability of satellite imagery for large scale applications. GEE is a web-based platform providing global-scale and analysis-ready earth observation data and the cloud computing infrastructure required for their analysis [5]. For SAR backscatter imagery, however, Analysis Ready Data (ARD) is a concept that doesn't have a fixed definition and can vary by application.

For example, radiometric terrain normalization is a recommended preprocessing step for most forest applications whereas, in ice monitoring, due to the absence of reliable

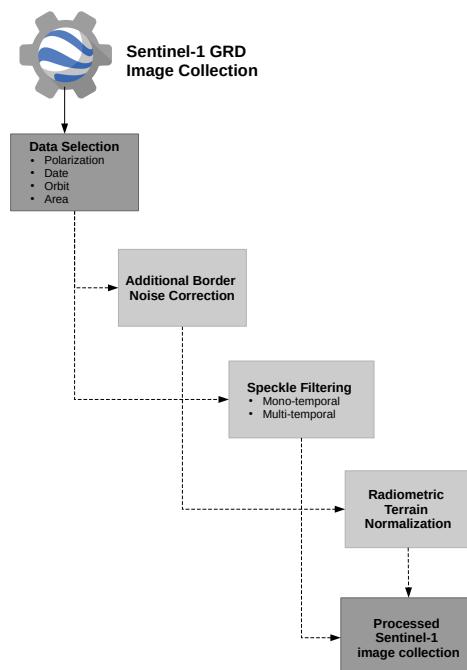
and up-to-date digital surface models, radiometric terrain normalization is usually not performed. To preserve the maximum information content and user freedom to adapt and optimize the preprocessing to specific application needs, the processing of the Sentinel-1 SAR backscatter data ingested in GEE is limited to thermal noise removals, data calibration, multi-looking, and range-doppler terrain correction, however, additional preprocessing is not performed.

Specifications and recommendations on additional preprocessing to generate SAR backscatter ARD are being developed [6,7]. The most authoritative guide towards SAR backscatter ARD comes from the Committee on Earth Observation Society (CEOS) Analysis-Ready-Data for Land (CARD4L) [7]. For a range of applications, CARD4L recommends additional preprocessing steps, such as speckle filtering and radiometric terrain normalization, to be implemented by the user to make the data ready for information extraction. However, due to the technical nature of SAR data, expert knowledge is needed to implement the additional preprocessing steps, which is still limiting the uptake of SAR data within the wider remote sensing community [8].

To support additional Sentinel-1 SAR data preprocessing in GEE, some implementations can be found scattered in online forums while other components are published. Here, we combine existing and new GEE implementations in a comprehensive framework for preparing Sentinel-1 SAR backscatter ARD in GEE suitable for a wide range land and inland water applications that includes additional border noise correction, speckle filtering and radiometric terrain normalization. The provided ARD preparation framework supports a wide range of monitoring and mapping applications.

## 2. Sentinel-1 SAR Backscatter ARD Preparation Framework

The implemented framework to prepare Sentinel-1 SAR backscatter ARD (Figure 1) includes additional border noise removal, speckle filtering, and radiometric terrain normalization applied to a Sentinel-1 GEE image collection. All the additional preprocessing is optional and can be adjusted by the user depending on their specific requirements and application needs. The framework is implemented in both GEE Javascript and the Python application programming interface (API) ([https://github.com/adugnag/gee\\_s1\\_ard](https://github.com/adugnag/gee_s1_ard)).

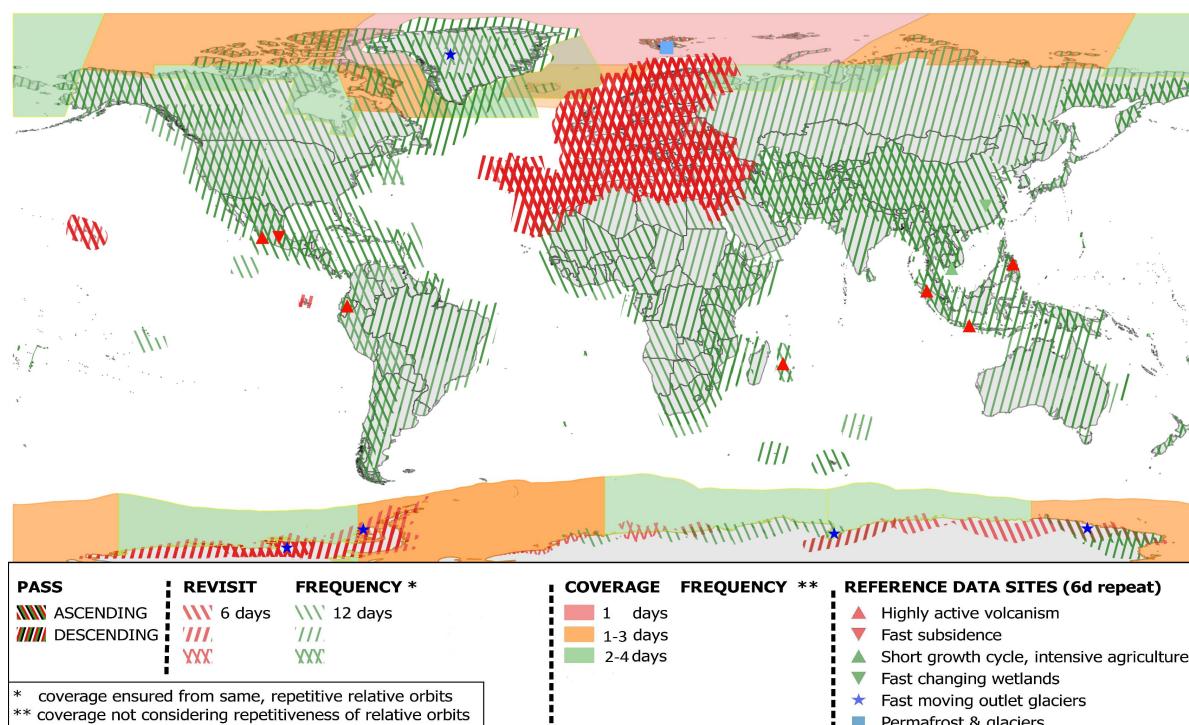


**Figure 1.** The preparation framework to create Sentinel-1 ARD in GEE. The dark grey box depicts the mandatory steps and the light grey ones depict the optional preprocessing steps.

## 2.1. Sentinel-1 Data Selection

The Sentinel-1 collection in GEE includes Sentinel-1 Ground Range Detected (GRD) scenes, processed using the Sentinel-1 Toolbox to generate a calibrated, ortho-corrected product. The collection includes all Sentinel 1-A and 1-B GRD image products acquired in both ascending and descending orbits and is updated daily. The GRD images utilized in this framework are acquired in interferometric wide swath (IW) mode. The Sentinel-1 GRD images have a pixel spacing of 10 m at which the full information detail is guaranteed, and a spatial resolution, i.e., the ability to separate between adjacent target objects on the ground of approximately  $20 \times 22$  m [9].

The Sentinel-1 constellation acquires images globally every 6 to 12 days, while, in some regions, it has a higher revisit time due to acquisition overlap (Figure 2) [10]. We used the Sentinel-1 GRD data in the GEE catalogue that are retained in linear scale (S1\_GRD\_FLOAT) as all the preprocessing steps require the data to be in linear scale. Data can be selected in single polarization i.e., 'VH' or 'VV' or dual polarization mode i.e., 'VV' and 'VH'.



**Figure 2.** Sentinel-1 constellation observation, revisit and coverage scenario (April 2021) [10].

We did not consider the Sentinel-1 imagery in 'HH' and 'HV' polarization as the coverage is predominantly in polar regions and our proposed preprocessing framework is designed for land and inland water application.

## 2.2. Additional Border Noise Correction

The presence of border noise in Sentinel-1 GRD products is one of the unwanted artifacts that should be removed before further analysis [11]. Originally, the border noise artifacts were produced when transforming the raw data to level-1 products by the Sentinel-1 Instrument Processing Facility software (IPF) 2.34 to 2.84 [11]. The dataset produced after upgrading the IPF to the new version released on March 2018 correctly applies the GRD border noise removal functionality. However, even after applying border noise correction, artifacts still remain in some images.

We implemented an additional masking procedure that removed most of the remaining border artifacts. The additional border noise removal followed in our implementation was modified from Hird et al. [12] (supplementary material) and is based on an incidence

angle threshold and removing anomalous backscatter values from the neighboring pixels in the border region. For details on border noise removal please refer to Stasolla and Neyt [11] and Hird et al. [12] (supplementary material).

### 2.3. Speckle Filtering

Speckle filtering is an important component of SAR image preprocessing as all SAR images are inherently affected by speckle. We have implemented key mono-temporal and multi-temporal speckle filters available in the literature. The mono-temporal speckle filters apply filtering to each individual image independently. These speckle filters range from a simple blind low-pass filter, such as the boxcar filter, to different adaptive filters, such as the Lee filter [13], Gamma Maximum A-posterior (MAP) Filter [14,15], Refined Lee filter [16], and the improved Lee sigma filter [17]. The Refined Lee speckle filter used in this note, was adopted from [18]. We also implemented the multi-temporal image filtering framework introduced by Quegan and Yu [19] in combination with the above mentioned mono-temporal filters.

To make multi-temporal filtering more consistent, we included additional parameters, such as the number of images used in the multi-temporal filtering framework. Additionally, only images with a similar geometry were considered in the multi-temporal filtering framework irrespective of the acquisition orbit in order to not degrade the quality of the speckle-filtered Sentinel-1 image collection. This was done by frame matching between images available in the collection to include only fully overlapping image frames.

Users can evaluate the performance of speckle filters qualitatively by visually checking the speckle reduction in homogeneous regions and the preservation of subtle features, such as roads and point scatterers, in the filtered images or quantitatively using the equivalent number of looks (ENL) or coefficient of variation in homogeneous regions in the image. Please refer to Di Martino et al. [20] and Mullissa et al. [21] for details.

### 2.4. Radiometric Terrain Normalization

Radiometric terrain normalization is important to mitigate the effect of topography on the SAR backscatter. We implemented a simplified angular-based correction approach suggested in Hoekman and Reiche [22]. Since the Sentinel-1 images ingested in GEE were in ground range and all orbital state vectors were lost, we used the GEE implementation of Vollrath et al. [8] that adopted the volume scattering and surface scattering models from Hoekman and Reiche [22]. Both reference models depend on the angular relationships between the SAR image and the terrain geometry. The angular based radiometric terrain normalization implemented in GEE depends only on angles, from which a simplified relation between the image and terrain can be derived.

By default, we used the Shuttle Radar Topography Mission (SRTM) digital elevation model (DEM) at a resolution of 1 arc-second ( $\sim 30$  m) [23], available as a GEE asset, to derive the elevation, slope and aspect values used in the radiometric terrain normalization. For areas greater than 60 degrees latitude where SRTM is not available, an alternative DEM, such as ASTER DEM [24], should be used. The SRTM DEM may also be substituted by regional high-resolution DEMs that better fit the spatial resolution of Sentinel-1, available as part of the Google Earth Engine collection or as uploaded by the user.

The radiometric terrain normalization provides a simplified mask to remove pixels in the active layover and shadow area in the image [8,22]. In this implementation, regions falling under an active layover and shadow areas are masked out by default. We also provide an optional additional buffer functionality set in 'meters' to account for passive layover and shadow in the image. Please refer to Vollrath et al. [8] for a detailed description.

### 2.5. Output

The processed Sentinel-1 ARD image collection can be formatted to a linear power or dB scale. For radiometrically terrain corrected outputs, the data will always be in gamma naught ( $\gamma^0$ ) as the terrain normalization in Hoekman and Reiche [22] is defined for  $\gamma^0$ .

In the absence of radiometric terrain normalization, the data is processed in sigma naught ( $\sigma^0$ ). To export the processed image collection, we provided the export-to-asset option so that every image in the collection can be exported to a GEE asset. Experienced users can, however, modify the script to export their data to Google Drive or Google Cloud storage.

The Sentinel-1 ARD preparation framework is implemented in both the JavaScript in the GEE code editor (Figure 3) and the GEE Python API ([https://github.com/adugnag/gee\\_s1\\_ard](https://github.com/adugnag/gee_s1_ard)). For quick prototyping and visualization, the JavaScript code editor is well suited, whereas, for large-scale operational processing, the Python API is appropriate. If the dual polarization state is selected, the first image taken from the processed collection is visualized in RGB ('VV', 'VH', 'VV/VH' ratio) (Figure 4). To aid in visualization and interpretation, the output data format can be changed to a dB scale by the user.

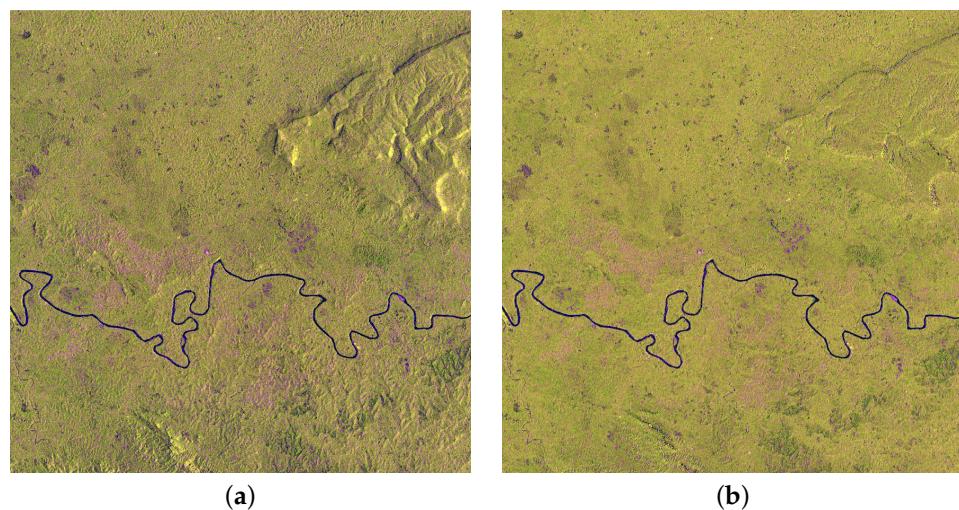
The screenshot shows the GEE JavaScript code editor interface. The top bar includes 'Get Link', 'Save', 'Run', 'Reset', and 'Apps'. The main area displays a script titled 's1.ard' with numerous lines of code related to Sentinel-1 ARD processing. The bottom half of the window shows a map of a coastal area with a black polygon outline indicating the region of interest. The map is overlaid with various colors representing different backscatter ratios. The right side of the interface has tabs for 'Inspector', 'Console', and 'Tasks', with several status messages listed under each.

```

s1.ard
65 //--- DEFINE PARAMETERS -----
66 //-----// 
67 //-----// 
68 //-----// 
69 //-----// 
70+ var parameter = { //1. Data Selection
71   START_DATE: '2018-07-05',
72   STOP_DATE: '2018-08-19',
73   POLARIZATION: 'VV',
74   ORBIT: 'DESCENDING',
75   //GEOMETRY: geometry, //uncomment if interactively selecting a region of interest
76   GEOMETRY: ee.Geometry.Polygon([[104.88, 11.61],[104.88, 11.36],[105.16, 11.36],[105.16, 11.16],[104.88, 11.16],[104.88, 11.61]]),
77   //ee.Geometry.Polygon([[112.05, -0.25],[112.05, -0.45],[112.25, -0.45],[112.25, -0.25],[112.05, -0.25]]),
78   //2. Additional Border noise correction
79   APPLY_ADDITIONAL_BORDER_NOISE_CORRECTION: true,
80   //3. Speckle filter
81   APPLY_SPECKLE_FILTERING: true,
82   SPECKLE_FILTER_FRAMEWORK: 'MULTI',
83   SPECKLE_FILTER: 'Lee',
84   SPECKLE_FILTER_KERNEL_SIZE: 9,
85   SPECKLE_FILTER_NR_OF_IMAGES: 10,
86   //4. Radiometric terrain normalization
87   APPLY_TERRAIN_FLATTENING: true,
88   DEM: ee.Image('USGS/SRTMGL1_093'),
89   TERRAIN_FLATTENING_MODEL: 'VOLUME',
90   TERRAIN_FLATTENING_ADDITIONAL_LAYEROVER_SHADOW_BUFFER: 0,
91   //5. Output
92   FORMAT: 'DB',
93   CLIP_TO_ROI: false,
94   SAVE_ASSETS: false
95 }
96

```

**Figure 3.** A screenshot of the GEE JavaScript code editor window.



**Figure 4.** RGB visualization of a dual polarized (VV and VH) Sentinel-1 SAR backscatter image of central Borneo, Indonesia (Lat:  $-0.35$ , Lon:  $112.15$ ) (a) as ingested into the Google Earth Engine; and (b) after applying additional border noise removal, a  $9 \times 9$  multi-temporal Gamma MAP speckle filter and radiometric terrain normalization with a volume scattering model. Here VV is in red, VH is in green, and VV/VH ratio is in blue.

### 3. Discussion

We proposed a framework to prepare Sentinel-1 SAR backscatter ARD in GEE that is applicable for a wide range of large-scale land and inland water applications. Having Sentinel-1 backscatter ARD available in GEE follows the philosophy of bringing algorithms to data, thus, eliminating the costly process of downloading and storing large amounts of data. While some of the methods described in this technical note have already been published and others have been implemented in GEE, the provision of a comprehensive framework combining processing steps now provides a tool for GEE-based large-area mapping and monitoring applications.

The implementation of speckle filters and radiometric terrain normalization in GEE gives the user options to use different parameters for preprocessing the Sentinel-1 images depending on the envisioned application and scale of processing. The proposed ARD framework's capability has been demonstrated for generating Sentinel-1-based forest disturbance alerts in the Congo basin, Africa [1] and in the Sepal platform (<https://sepal.io>) for preprocessing Sentinel-1 SAR data.

### 4. Conclusions

We provided an implementation both in GEE JavaScript and Python API to achieve Sentinel-1 SAR backscatter ARD that is ready for information extraction. We believe these implementations will make preprocessing Sentinel-1 images in GEE easier for a wider range of researchers and non-experts.

**Author Contributions:** Conceptualization, A.M., J.R., and A.V.; Implementation of algorithm A.M., C.O.-B., and A.V.; Testing and bug reporting N.G., B.S., J.B., Y.G., and J.R.; writing—original draft preparation, A.M.; editing, A.V., C.O.-B., N.G., J.B., Y.G., J.R., and B.S.; funding acquisition, J.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was partly funded through the U.S. government SilvaCarbon Program.

**Data Availability Statement:** The codes are openly available from [https://github.com/adugnag/gee\\_s1\\_ard](https://github.com/adugnag/gee_s1_ard).

**Acknowledgments:** We are grateful for the valuable comments provided on the manuscript by Nicholas Clinton of Google. We thank the two anonymous reviewers for their comments that helped to further improve the manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Reiche, J.; Mullissa, A.; Slagter, B.; Gou, Y.; Tsendlbazar, N.E.; Odongo-Braun, C.; Vollrath, A.; Weisse, M.J.; Stolle, F.; Pickens, A.; et al. Forest disturbance alerts for the Congo Basin using Sentinel-1. *Environ. Res. Lett.* **2021**, *16*, 024005. [[CrossRef](#)]
2. Nguyen, D.B.; Wagner, W. European rice cropland mapping with Sentinel-1 data: The Mediterranean region case study. *Water* **2017**, *9*, 392. [[CrossRef](#)]
3. Lanari, R.; Bonano, M.; Casu, F.; Luca, C.D.; Manunta, M.; Manzo, M.; Onorato, G.; Zinno, I. Automatic generation of sentinel-1 continental scale DInSAR deformation time series through an extended P-SBAS processing pipeline in a cloud computing environment. *Remote Sens.* **2020**, *12*, 2961. [[CrossRef](#)]
4. Pekel, J.F.; Cottam, A.; Gorelick, N.; Belward, A.S. High-resolution mapping of global surface water and its long-term changes. *Nature* **2016**, *540*, 418–422. [[CrossRef](#)] [[PubMed](#)]
5. Gorelick, N.; Hancher, M.; Dixon, M.; Ilyushchenko, S.; Thau, D.; Moore, R. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sens. Environ.* **2017**, *202*, 18–27. [[CrossRef](#)]
6. Truckenbrodt, J.; Freemantle, T.; Williams, C.; Jones, T.; Small, D.; Dubois, C.; Thiel, C.; Rossi, C.; Syriou, A.; Giuliani, G. Towards Sentinel-1 SAR analysis-ready data: A best practices assessment on preparing backscatter data for the cube. *Data* **2019**, *4*, 93. [[CrossRef](#)]
7. Lewis, A.; Lace, J.; Mecklenburg, S.; Ross, J.; Siqueira, A.; Killough, B.; Szantoi, Z.; Tadono, T.; Rosenqvist, A.; Goryl, P.; et al. CEOS analysis ready data for Land (CARD4L) overview. In Proceedings of the IGARSS 2018–2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 7407–7410.
8. Vollrath, A.; Mullissa, A.; Reiche, J. Angular-based radiometric slope correction for Sentinel-1 on google earth engine. *Remote Sens.* **2020**, *12*, 1867. [[CrossRef](#)]

9. Torres, R.; Snoeij, P.; Geudtner, D.; Bibby, D.; Davidson, M.; Attema, E.; Potin, P.; Rommen, B.; Flory, N.; Brown, M.; et al. GMES Sentinel-1 mission. *Remote Sens. Environ.* **2012**, *120*, 9–24. [[CrossRef](#)]
10. ESA. Sentinel-1 Observation Scenario. Available online: <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-1/observation-scenario> (accessed on 13 April 2021).
11. Stasolla, M.; Neyt, X. An Operational Tool for the Automatic Detection and Removal of Border Noise in Sentinel-1 GRD Products. *Sensors* **2018**, *18*, 3454. [[CrossRef](#)] [[PubMed](#)]
12. Hird, J.N.; DeLancey, E.R.; McDermid, G.J.; Kariyeva, J. Google earth engine, open-access satellite data, and machine learning in support of large-area probabilistic wetland mapping. *Remote Sens.* **2017**, *9*, 1315. [[CrossRef](#)]
13. Lee, J.S. Digital image enhancement and noise filtering by use of local statistics. *IEEE Trans. Pattern Anal. Mach. Intell.* **1980**, *2*, 165–168. [[CrossRef](#)] [[PubMed](#)]
14. Lopes, A.; Nezry, E.; Touzi, R.; Laur, H. Maximum a posteriori speckle filtering and first order texture models in SAR images. In Proceedings of the 10th Annual International Symposium on Geoscience and Remote Sensing, College Park, MD, USA, 20–24 May 1990; pp. 2409–2412.
15. Lopes, A.; Touzi, R.; Nezry, E. Adaptive speckle filters and scene heterogeneity. *IEEE Trans. Geosci. Remote Sens.* **1990**, *28*, 992–1000. [[CrossRef](#)]
16. Lee, J.S.; Grunes, M.R.; De Grandi, G. Polarimetric SAR speckle filtering and its implication for classification. *IEEE Trans. Geosci. Remote Sens.* **1999**, *37*, 2363–2373.
17. Lee, J.S.; Wen, J.H.; Ainsworth, T.L.; Chen, K.S.; Chen, A.J. Improved sigma filter for speckle filtering of SAR imagery. *IEEE Trans. Geosci. Remote Sens.* **2008**, *47*, 202–213.
18. Lemoine, G. Refined Lee GEE Implementation. Available online: <https://code.earthengine.google.com/5d1ed0a0f0417f098fdfd2fa137c3d0c> (accessed on 1 April 2021).
19. Quegan, S.; Yu, J.J. Filtering of multichannel SAR images. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 2373–2379. [[CrossRef](#)]
20. Di Martino, G.; Poderico, M.; Poggi, G.; Riccio, D.; Verdoliva, L. Benchmarking framework for SAR despeckling. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 1596–1615. [[CrossRef](#)]
21. Mullissa, A.G.; Marcos, D.; Tuia, D.; Herold, M.; Reiche, J. deSpeckNet: Generalizing Deep Learning-Based SAR Image Despeckling. *IEEE Trans. Geosci. Remote Sens.* **2020**. [[CrossRef](#)]
22. Hoekman, D.H.; Reiche, J. Multi-model radiometric slope correction of SAR images of complex terrain using a two-stage semi-empirical approach. *Remote Sens. Environ.* **2015**, *156*, 1–10. [[CrossRef](#)]
23. Farr, T.G.; Rosen, P.A.; Caro, E.; Crippen, R.; Duren, R.; Hensley, S.; Kobrick, M.; Paller, M.; Rodriguez, E.; Roth, L.; et al. The shuttle radar topography mission. *Rev. Geophys.* **2007**, *45*. [[CrossRef](#)]
24. Abrams, M.; Crippen, R.; Fujisada, H. ASTER Global Digital Elevation Model (GDEM) and ASTER Global Water Body Dataset (ASTWBD). *Remote Sens.* **2020**, *12*, 1156. [[CrossRef](#)]