SoftwareX 26 (2024) 101717

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

Floodsar: Automatic mapping of river flooding extent from multitemporal SAR imagery

Tomasz Berezowski *, Szymon Niemiec, Andrzej Chybicki

Gdańsk University of Technology, Faculty of Electronics, Telecommunication and Informatics, ul. Gabriela Narutowicza 11/12 80-233 Gdańsk, Poland

ARTICLE INFO

Dataset link: Input files for the Floodsar software (Original data)

Keywords:
Flood extent
SAR
Thresholding
Classification
Sentinel-1

ABSTRACT

Floodsar is an open-source tool for automatic mapping of the flood extent from a time series of synthetic aperture radar (SAR) imagery. Floodsar is unsupervised, however, it requires defining the parameters search space, geographical area of interest, and some river gauge observations (e.g. water levels or discharges) time series that overlap temporarily with the SAR imagery. Applications of Floodsar are mainly in real-time monitoring and elaborating long-time series of historical data. Floodsar features two algorithms for flood extent mapping. The 1D algorithm identifies a flood/no-flood threshold in one SAR polarization. The 2D algorithm performs clustering on two SAR polarizations at the same time. Floodsar chooses the optimal threshold or the composition of clusters by maximizing the correlation between the resulting flood area and the river gauge observations. Floodsar was tested on three case studies with different land uses to illustrate its performance. The 2D algorithm performed on average the best with the average kappa=0.78, yet the less complex 1D/VV algorithm obtained similar results.

Code metadata

Current code version	v1.2
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00787
Permanent link to reproducible capsule	–
Legal code license	MIT License
Code versioning system used	git
Software code languages, tools and services used	C++ 17, GDAL
Compilation requirements, operating environments and dependencies	OS: Linux, Windows Subsystem for Linux (WSL). Dependencies: build-essential, g++, cmake, ninja-build, libgdal-dev, gdal-bin
If available, link to developer documentation/manual	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00787
Support email for questions	tomberez@eti.pg.edu.pl

1. Motivation and significance

Floodsar is a software that performs preprocessing of multitemporal synthetic aperture radar (SAR) imagery and analyzes the imagery in the scope of relation with river gauge observations in order to map the flooding extent. The Floodsar software allows to identify flood extent, or absence of flood, in nearly real-time without supervision upon proper initial setup. Primarily, it aims to provide a long time series of flood extent, which can either be used for calibration and validation of hydrological models, or for ecological and vulnerability research. Due to the near-real-time operation, easy integration with a cloud, and ability to identify a flooding extent at a considerable distance from a river gauge (but within the same river system), Floodsar can also be used in crisis management.

Until now, one of the algorithms used in Floodsar was used to produce flood extent maps for the Biebrza wetlands, Poland [1]. As shown in that study, the flooding frequency map produced for the 2014–2019 period using 161 SAR imagery resembled well the flood frequency maps obtained in other studies using a hydrodynamic model. Further, these flood extent maps were used to validate a groundwater/surface water hydrological model, which was used to investigate the spatially variable

* Corresponding author.
E-mail address: tomberez@eti.pg.edu.pl (Tomasz Berezowski).

https://doi.org/10.1016/j.softx.2024.101717
Received 20 November 2023; Received in revised form 28 February 2024; Accepted 25 March 2024
Available online 5 April 2024

extent of the river, groundwater, rain, and snow water in the flood for the past and future (1881–2099) climate [2].

Works on remote methods for mapping flood extent have been primarily using optical, or visible and infrared (VIS-IR) light sensors and SAR. While spectral features extracted from VIS-IR imagery effectively discriminates water from the surrounding land cover, the key advantage of SAR sensors is their ability to record flood conditions irrespective of weather conditions or solar illumination. Although VIS-IR methods, such as the modified normalized difference water index (MNDWI, [3]) thresholding is efficient for flood mapping [4], their operational applicability is limited due to reliance on atmospheric conditions. This limitation is particularly significant for real-time flood disaster monitoring, short-term forecasting [5], and acquisition of continuous time series for scientific analyses.

In the past two decades, methods using digital analysis of SAR observation, such as threshold methods [6,7], change detection [8,9], or classification [10,11] a have become commonplace. In recent years, increasing progress in this area has come from machine learning tools such as Random Forest [12], convolution neural networks [13], unsupervised [14], or hybrid [15].

Despite numerous flood classification approaches for remote sensing data, only a few have been implemented in operational systems with accessible data. The Global Flood Database [16] utilized data from two MODIS sensors and ancillary elevation data to generate 250 m resolution flooding extents from 2000 to 2018 [17]. The workflow in the Global Flood Database involves automatic thresholding on multitemporal reflectance bands and band ratios, followed by rule-based detection and elevation-slope-based masking to eliminate terrain shadow. The NOAA Global Flood Product [18] provides near real-time 375 m resolution data using VIIRS sensors, along with other VIS-IR sensors and ancillary data, including a digital elevation model [19,20]. The algorithm behind the Global Flood Product requires multiple processing steps such as shadow removal, water detection, minor flood detection, and water fraction retrieval to determine floods. Hasard [21] is an on-demand solution that utilizes a Sentinel-1 SAR archive to provide data at approximately 20 m spatial resolution and location-dependent temporal resolution daily to sub-monthly. Hasard is a proprietary algorithm based on [22], which employs a hierarchical split-base approach to select tiles of a SAR image or of a change detection (flood/no flood) SAR image with bimodal histograms. The selected image tiles undergo processing with thresholding and region growth techniques to detect floods. The limitations of the first two approaches include a complex workflow with numerous ancillary data requirements, low spatial resolution, and missing data due to cloud cover. In the case of Hasard, these limitations do not apply; however, it is not an open-source solution. Consequently, a ready-to-use, open-source solution applicable to a specified location for creating a time series of data for scientific analysis is currently unavailable. The software presented herein, Floodsar, offers an original approach to overcoming the aforementioned limitations while maintaining similar features.

2. Software description

2.1. Software architecture

Floodsar is composed of three main components: (1) data preprocessing, (2) flood extent mapping, and (3) results postprocessing. The two former components are implemented in the “floodsar” and the latter component is implemented in the “mapper” executable file. The tool is available as source code and requires compilation. Floodsar requires only easily accessible input data: time series of SAR imagery, time series of river gauge observations such as water levels or discharge, and the area of interest (AOI). As mentioned earlier, two algorithms for automatic mapping of the flooding extent are available. The first one is the implementation of a thresholding method [1], and the other is a new algorithm based on clustering. The tool was developed in C++ 17, with the spatial data processing implemented using the Geospatial Data Abstraction Library (GDAL).

2.2. Software functionalities

2.2.1. Data preprocessing

Floodsar is designed to work with dual-pol Sentinel-1 data in the interferometric wide (IW) swath mode in the ground range detected (GRD) processing level [23]. Both VV (vertical/vertical) and VH (vertical/horizontal) polarizations are used for flood mapping. The algorithm can also process data from other sensors as long as two polarizations are provided, however, due to high temporal resolution, currently only data from the Sentinel-1 constellation is suitable for easy use in Floodsar. The input data parser is designed to automatically recognize Sentinel-1 imagery prepossessed by the Alaska Satellite Facility (ASF) on-demand Hybrid Pluggable Processing Pipeline (HyP3) [24]. The integration with HyP3 output decreases the workload for the user as all essential SAR processing steps, such as identification of the precise orbit, radiometric calibration, speckle filtering, and geometric corrections are done automatically by a remote machine. Alternatively, a user can conduct their own SAR processing using, for example, the European Space Agency (ESA) SNAP software. However, it is essential to properly name the output files so that they can be recognized by Floodsar.

In the first step of preprocessing, the names of all SAR raster files are parsed to identify the acquisition date and polarization. Subsequently, the imagery is re-projected into a coordinate system selected by the user. This re-projection is necessary because Sentinel-1 imagery from different relative orbits often utilizes varying coordinate systems, such as consecutive zones of the Universal Transverse Mercator coordinate system. If multiple images with the same date and polarization are found, they are merged into a single, continuous raster (mosaic). Following this, the imagery is cropped to the extent of the Area of Interest (AOI), and the pixel values are preserved for use in the subsequent flood extent mapping step.

Floodsar requires river water level [L] or discharge [L^3T^{-1}] time series input data (hereinafter referred to as river gauge observations). The river gauge observations have to overlap temporarily with the SAR imagery time series. More river gauge observations than SAR images can be provided, as the tool will only select those observations that correspond to the dates of the SAR imagery.

2.2.2. The 1D algorithm for flood mapping

The first algorithm used for flood extent mapping uses only one SAR polarization at a time, hence it was named the 1D algorithm. The 1D algorithm is an implementation of the automatic threshold identification method by [1]. The assumption behind the method is that river gauge observations are proportional to the flooding extent area visible in SAR imagery. The flooding extent is defined as SAR imagery pixels with the backscatter coefficient b in VV or VH polarization below the t_0 threshold. The b units can be either linear power or dB. Both backscatter not normalized and normalized to the incidence angle, i.e. σ^0 and γ^0, can be used as b. The t_0 threshold is estimated by maximizing the Pearson’s correlation between a time series of a n day river gauge observations $H = (h_1, h_2, \ldots, h_n)$ and a corresponding time series of flooding extent areas from binarized SAR imagery:

$$U(t) = \left(u_1(t), u_2(t), \ldots, u_n(t)\right) \tag{1}$$

$$u_i(t) = \sum_{j=1}^{m} \begin{cases} a, & b_{ij} \le t \\ 0, & b_{ij} > t \end{cases} \tag{2}$$

where t is a binarization threshold for all images, i identifies a single SAR imagery 1 to n, j identifies a SAR imagery pixel from 1 to m, and a is a pixel area [L^2]. Given the search space $t \in [t_{min}, t_{max}]$, the optimal threshold is estimated as:

$$t_0 = \underset{t \in [t_{min}, t_{max}]}{\arg\max}\ \mathrm{cor}\left(U(t), H\right) \tag{3}$$

The 1D algorithm is comparable to the frequently used Otsu [25] method because the differentiation between flooded and not-flooded

areas is determined by a threshold value. Contrary to the Otsu [25] method, the difference in the 1D algorithm is in the physically-based assumption and applicability to multitemporal imagery. As illustrated in [1] and in the case study examples in Section 3, the 1D algorithm is very robust in terms of capturing the flooding extent. The advantage of the 1D algorithm is that it requires no parameters other than the definition of the search space, which must fall within the range of expected floodwater backscatter values found in the imagery. The major limitation of this algorithm is that the area of the flooding extent has to be positively correlated to the river gauge observations. This may not always be the case when a river bank or a floodplain margin has a steep slope (within one SAR pixel) effectively producing the same water extent for a range of water levels (e.g. when dikes are present). Also, Floodsar will not identify the relationship between flood area and river gauge observations when water is within a narrow (close to the SAR imagery resolution) riverbed due to the low variability of the water extent in the SAR imagery. The flat land surface features like river ice, bare soil, etc. will also affect the functioning of the 1D algorithm. Although these features are not correlated to the river gauge observations, they may influence the estimated t_0 value. Moreover, these features will be labeled as flooded in the 1D algorithm output if their backscatter is lower than t_0. The conceptual limitation of the 1D algorithm is that it cannot make use of both polarizations at the same time.

2.2.3. The 2D algorithm for flood mapping

The second algorithm used for flood extent mapping uses both VV and VH SAR polarizations at a time, hence it was named the 2D algorithm. Unlike the 1D algorithm, the 2D algorithm was developed specifically for Floodsar. Similarly to the 1D algorithm, the 2D algorithm identifies flooding extent based on correlation with river gauge observation, but the flooding extent mapping is performed by selecting a set of flood/no-flood classes from unsupervised clustering output.

The 2D algorithm requires providing n SAR imagery in VV and VH polarization and a temporarily overlapping time series of river gauge observations (H). The method consists of two major steps: clustering and optimal cluster selection (Listing 1). First, all pixels from the imagery are transformed into a two-dimensional matrix, where the first dimension represents a flattened spatial and temporal dimension of the imagery and the second dimension represents the polarization. Then, the k-means [26] clustering is performed on this matrix for each k [-] target class number in the $[k_{min}, k_{max}]$ range, and the resulting k centroids are saved for the next step. The k_{min} and k_{max} are input parameters with $k_{min} \geq 2$. Next, for each clustering result, flood class candidate mapping is performed sequentially for f sets of classes in the $[1, k-1]$ range, where f represents the number of flooded classes, and the remaining classes are considered as non-flooded. The strategy of selecting classes for subsequent f sets is based on increasing values of the class centroids, i.e. the class with the lowest centroid values is selected for $f = 1$. The rationale behind this order of class selection is that the water extent has the lowest backscattering coefficient in the SAR image. After mapping the flood class candidates, the flooded pixels in every n imagery are summed and the Pearson's correlation between the river gauge observations, H, and flood areas in each f and k combination is calculated. Finally, the optimal f and k are selected based on the highest correlation as f_{opt} and k_{opt}. In summary, the 2D algorithm is an extension of unsupervised clustering (k-means in our case) to automatic class labeling and the k meta-parameter optimization.

The multidimensional clustering in the 2D algorithm overcomes the conceptual limitation of the 1D algorithm, i.e., using only one SAR polarization. Therefore, the decision, of whether to label a certain location as flooded or not flooded is based on more data. Yet, the disadvantage of the 2D algorithm is that the results are sensitive to the $[k_{min}, k_{max}]$ range. This usually requires setting a broad $[k_{min}, k_{max}]$ range, which considerably increases the computation time in reference to the 1D algorithm. A solution to this could be using a Floodsar option which allows to estimate the cluster centroids based on a random fraction of pixels given that the fraction is not too small and the AOI is representative for the flood dynamics.

2.2.4. Results post-processing and the cache

The output of processing steps is stored in the cache. Therefore time-consuming steps, such as SAR image mosaicking, clipping, and reprojecting or k-means clustering does not have to be repeated if one wants to test Floodsar with the same data set, but different parameters. If one decides that the cache should not be used then it is erased at the beginning of a new run of application.

The cached data is also used by the "mapper" executable for post-processing the SAR images into flood extent raster maps. The "mapper" binary automatically chooses the best thresholding or clustering parameters produced by "floodsar". However, one can analyze results from the entire search space $[k_{min}, k_{max}]$ if an appropriate option is set during the "mapper" run. After the post-processing step, the flood maps for each input SAR imagery date are saved to individual GeoTIFF files in the coordinate system as specified during the Floodsar run.

input : VV, VH as n dual-pol SAR imagery
input : H as n river gauge observations
input : k_{min}, k_{max} as a range of k-means target clusters to search
output: k_{opt} and f_{opt} as an opidentifiedtimal numbers of the total k-means clusters and the flood clusters

Pixels = two column empty matrix;
DateIndices = empty list of vectors;
`// all VV and VH data are stored in a two colum matrix`
for date $i = 1$ **to** n **do**
    Pixels[1].`append(flatten(`VV_i`))`;
    Pixels[2].`append(flatten(`VH_i`))`;
    DateIndices[i] = the Pixel matrix indices corresponding to the date i;
end
Centroids = empty list of vectors;
CorMatrix = empty matrix;
for $k = k_{min}$ **to** k_{max} **do**
    Centroids = `kmeans(`Pixels,k`)`;
    for $f = 1$ **to** $k - 1$ **do**
        `// store flood areas for` n `days given` k `and` f
        FloodArea = n element vector with zeros;
        for date $i = 1$ **to** n **do**
            `// mapping of VV and VH pixels to cluster IDs given the k-means centroids`
            FloodMap = `kmeansMap(`Centroids, Pixels[1:2, DateIndices[i]]`)`;
            foreach p in floodMap **do**
                FloodID = a set of f cluster IDs with lowest centroid values **if** $p \in$ FloodID **then**
                    FloodArea[i]++;
                end
            end
        end
        CorMatrix[f, k] = `correlation(`FloodArea, H`)`
    end
end
return Optimal k_{opt}, f_{opt} as indices of the CorMatrix where: CorMatrix == `max` (CorMatrix)

Listing 1: Pseudocode of the 2D algorithm.

3. Illustrative examples

To illustrate the Floodsar performance we have chosen three case studies varied by climate, land use, and character of flooding. The first

case is a series of two floods that occurred in California, USA, in the period December 2022 to March 2023 (32 images). There, we have chosen the section of Sacramento River located east of Sacramento city (121.6°W, 38.6°N). The climate is warm-summer Mediterranean and land-use is urban and agricultural. The second case is a flood that occurred southwest of York, UK, in the period December 2015 to January 2016 (43 images). There, we have chosen a location near the confluence of Ouse River and Wharfe River (1.1°W, 53.8°N). The climate is oceanic and land-use is predominantly agricultural land with small cities and an airport. The third case is the Biebrza River, where flooding occurs each year and is desired to sustain the ecology of the area. We have chosen the spring flooding event that occurred in 2017 (37 images). Out of the entire floodplain we have chosen a section upstream the outlet of Biebrza River (22.5°E, 53.3°N). The climate is humid continental and the land-use is a natural riverine fen wetland with small villages and agricultural fields located outside the floodplain. The flooding event in the Biebrza site is the only one that was triggered directly by snowmelt rather than by rainfall.

3.1. Data

We used the Hyp3 on-demand processing feature provided by ASF to obtain Sentinel-1A/B 10 m VV and VH polarization images with speckle filtering, radiometric calibration to γ^0 in linear power, pixel spacing of 10 m, and geometric calibration with digital elevation model matching (See details in [27]). We used imagery from both ascending and descending orbits, which were not limited to any particular orbit number.

We used various river gauge observations provided by the respective national agencies in the daily resolution. For the Sacramento case, we used discharge in cubic feet per second in a gauge located about 45 km downstream the Sacramento city [28], for the Ouse case, we used discharge in cubic meters per second for a gauge located at the Ouse River about 20 km upstream of the Ouse and Wharfe confluence [29]. For the Biebrza case, we used local water height in centimeters for a station located directly in the study area [30].

To verify the Floodsar results against independent data sources we used a MNDWI thresholding as in [31] for cloud-free images of Sentinel-2A/B satellite sensor during the extensive flooding for each site. Based on the MNDWI flood maps and the Floodsar results we have calculated the (1) kappa index, which quantifies agreement between these data sets with the maximum kappa=1 indicating perfect agreement and kappa≤0 for no agreement, (2) precision is the fraction of Floodsar flood pixels among all flooded pixels in an MNDWI flood map, and (3) sensitivity is the fraction of Floodsar flood pixels that agree with MNDWI flooded pixels among all Floodsar flood pixels.

3.2. Floodsar mapping results and discussion

The correlation between river gauge observations and flooding extents vary depending on the algorithm and SAR polarization used (Table 1). The correlation between flooding areas identified by Floodsar and river gauge observation was the highest for the 2D algorithm in Biebrza and Ouse cases and for the 1D algorithm with the VH data in the Sacramento case. In each study site, the 2D algorithm optimized to more than one flood cluster, however, in the Floodsar runs where the search space was limited to a smaller number of all clusters ($k_{max} \leq 4$), the 2D algorithm identified one flood cluster as optimal (result not shown). Needless to say, our preliminary test showed that both 1D and 2D algorithms can run with a much lower number of satellite images than used in this study. For example, in the Ouse case, Floodsar identified the flood extent correctly with five SAR images, which were, however specially selected to reflect the water level variability.

It is worth mentioning, that the correlation between flooding area and river gauge observations (Table 1) cannot be interpreted as the quality of flood mapping. This is because Floodsar does not use an absolute flood area, but its variance to determine the correlation. As long as the variance of time-dependent flooding area will match the variance of river gauge time-series the correlation will be strong. This is the best highlighted by the 1D/VH results for the Sacramento case, which has a very high correlation, but the optimized threshold is two orders of magnitude lower than for the remaining cases. Effectively the 1D/VH flooding area for Sacramento is much underestimated (cf. Table 1 and Fig. 1).

Table 1
The VV and VH polarization γ^0 thresholds t_0 (in linear power) optimized using the 1D algorithm and the optimal number of flood clusters f_{opt} accompanied by the total number of clusters k_{opt} optimized using the 2D algorithms. The optimal clusters and thresholds are accompanied by the correlation coefficient (cor.) calculated between the flooding areas and the river gauge observations.

Case study	1D				2D	
	VV		VH		f_{opt}/k_{opt}	cor.
	t_0	cor.	t_0	cor.		
Biebrza	0.0256	0.74	0.0068	0.77	4/11	0.78
Sacramento	0.02197	0.66	3.00E−05	0.75	4/10	0.69
Ouse	0.0313	0.61	0.0024	0.64	2/11	0.65

The 2D algorithm and 1D/VV algorithm results matched the total flood area from the reference flood maps, with the values being underestimated by 2% to 8% of the total area in Ouse and Sacramento cases (Fig. 1). The 1D VH results matched the reference flood area only for the Biebrza case. Although the 1D/VH algorithm results captured well the temporal variation of the flood area in the Sacramento and Ouse cases (see Table 1), the magnitude of the flood area was underestimated by up to 48% of the total area (Fig. 1). Overall, the 2D and 1D/VV algorithms performed very similar one to another in the Ouse and Sacramento cases while the only difference between the two was in the short-term variability of the flood area. The resulting flood area was smoother for the 2D algorithm compared to the 1D/VV algorithm. In the Biebrza case the 1D VV algorithm underestimated the total flood area by 8% while the 1D/VH and 2D algorithm results matched the flooding area better with 4% and 5% underestimation respectively.

Overall, the highest agreement of Floodsar results with the reference flood extent maps is observed for the 2D algorithm, which achieved an average kappa index of 0.780 across three case studies (Fig. 2). However, in the Sacramento case, the Kappa index for the 1D algorithm using VV data is 0.022 higher than for the 2D algorithm. The average precision across all case studies is greatest for the 1D/VV results (kappa index of 0.916), which was only marginally higher (by 0.005) than that for the 2D algorithm. The highest mean sensitivity across all case studies is observed for the 1D/VH results (0.997), surpassing the 2D algorithm by 0.008 and the 1D algorithm with VV data by 0.013. In summary, the pattern of the validation figures shows that the 1D/VV and 2D algorithm performed similarly for the Ouse and Sacramento cases while for the Biebrza case, the 2D algorithm performed better in terms of the kappa index.

The spatial pattern is similar for the Ouse and Sacramento cases with the most continuous flooding extent and the most false positives in the 1D/VV algorithm, the least continuous flooding extent with the least false positives in the 1D/ VH algorithm, and the 2D algorithm results in between the two latter (Fig. 3b and c). In the case of Biebrza (Fig. 3a) the 2D algorithm results exhibited less noise and discontinuities than both 1D/VV and VH results. Also in the Biebrza case Floodsar failed to identify a water patch visible in the MNDWI image (Figure 3a, Easting=2.2 km, Northing=4 km). This was the most likely due to dry vegetation remaining above the water surface, what increased the backscatter, but had a relatively low influence on the reflectance observed from near-nadir.

The different level of false positive detection is observable in the airport located in the southeastern part of the Ouse case (Fig. 3c) . The airport and surrounding flat areas of false positives comprise more

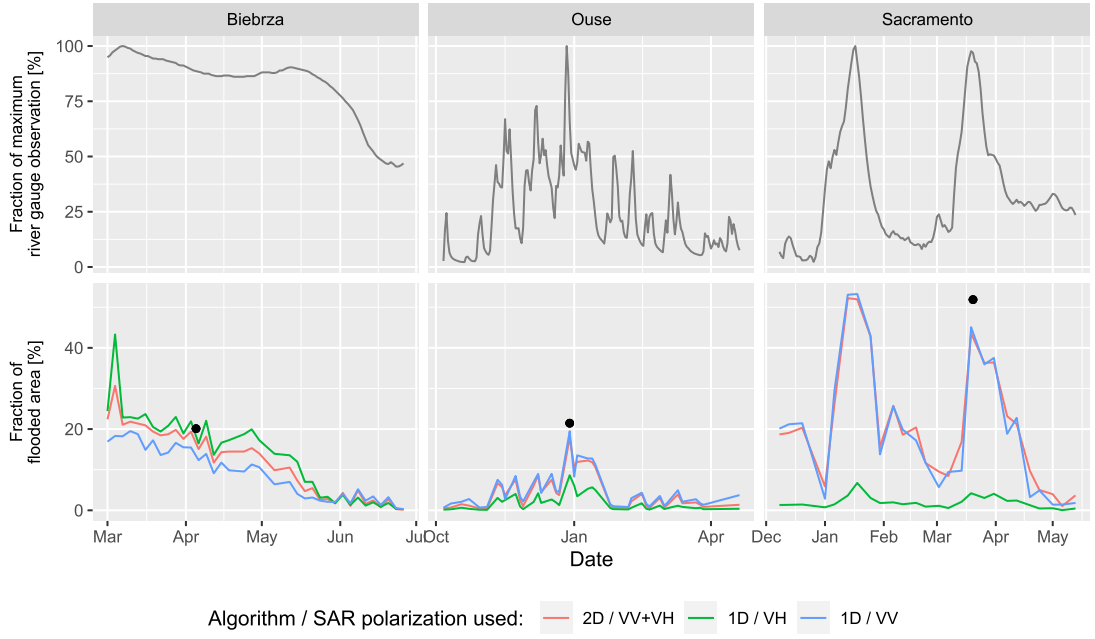


Fig. 1. Normalized river gauge observations in daily resolution (top panels) and normalized flooded area (bottom panels) identified by 1D and 2D algorithms of Floodsar in 2-12 days (2-3 days in majority) temporal resolution with the area from the reference flood maps indicated as black dots.

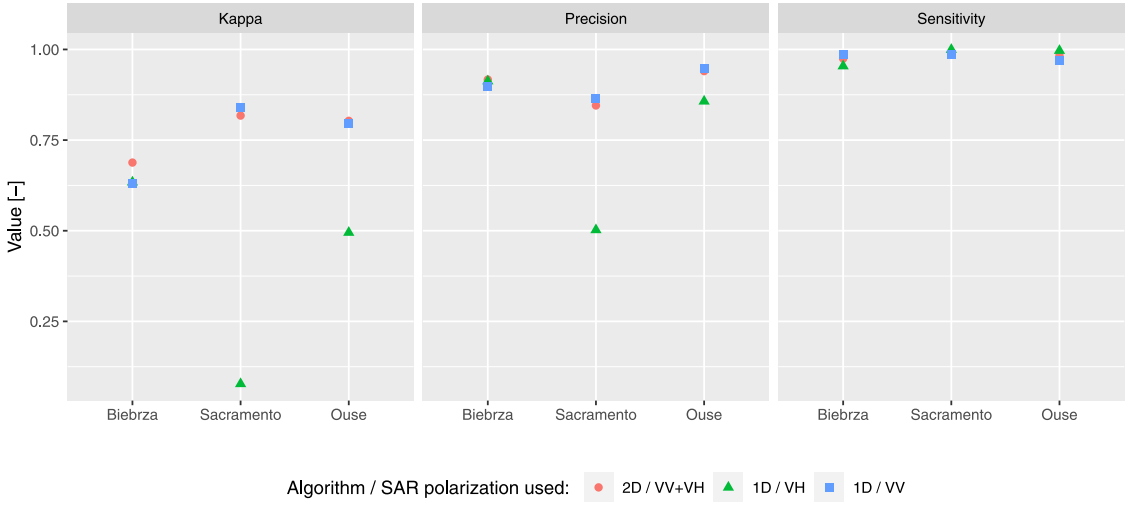


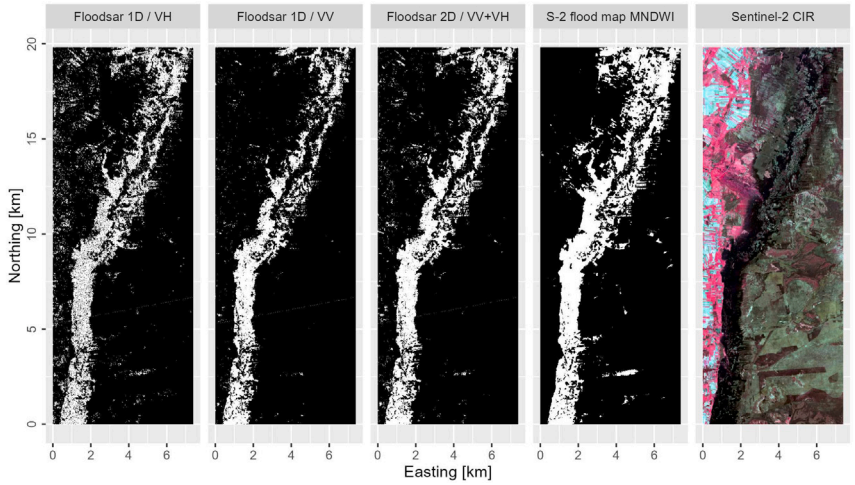
Fig. 2. The agreement of Floodsar results with the flood reference maps calculated using MNDWI thresholding. Kappa index, precision, and sensitivity were calculated for the reference flood map dates, as indicated in Fig. 1.

area in the 1D/VV and 2D results than in 1D/VH, where only isolated sections of the tarmac were labeled as flooded.

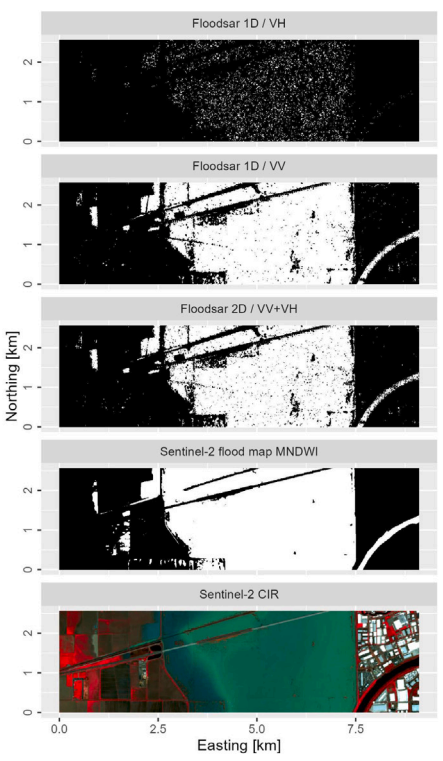
Lack of continuity in the 1D/VH flooding extent is visible the most in the Sacramento case (Fig. 3b), where a large part of the floodplain was sparsely covered by dry vegetation. Contrary to the Biebrza and Ouse cases, where the vegetation cover was dense, the VH backscatter in the Sacramento case was much weaker, similar to the VH backscatter of the open water.

Similar results obtained for the 1D/VV and 2D algorithms suggest that the two-dimensional algorithm regresses to one dimension, namely VV. This phenomenon may occur in sites where VH backscatter is too low to accurately identify floods, or where other objects in the study exhibit VH backscatter similar to open water, such as in the Ouse and Sacramento cases. However, as illustrated in the Biebrza case, VH backscatter carries information that is not redundant to VV and proves valuable in identifying flood extent (Fig. 4). In the Biebrza case, the 2D algorithm distinctly defines a flood/no flood boundary that neither aligns with the VV nor VH threshold values. The flood/no flood boundary of the 2D algorithm is in some sections almost perpendicular to the first principal component (PC) axis. This observation suggests that dimension reduction techniques with 1D thresholding could potentially be employed instead of the 2D algorithm. Nevertheless, the use of the 2D algorithm is justified in case studies like Biebrza because the 2D algorithm's boundary is not entirely perpendicular to PC1. Additionally, PC2 explains 13% of the variance with clear variability within the flood clusters.

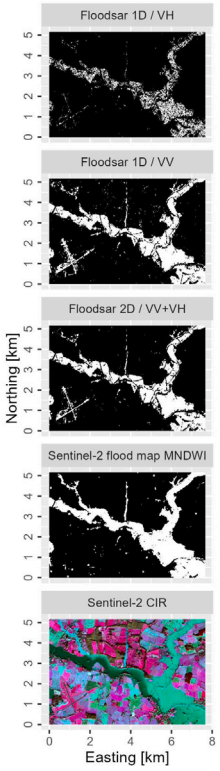
Floodsar does not discriminate between origins of open water, therefore initial water in the study area, permanent water, and time-variable flood water were identified in the case studies. This is well illustrated in the Sacramento case study, where at the beginning of the analysis discharge was lower and flooded area was higher than at the end of the analysis period. This is due to a large amount of initial water storage at the beginning of the analysis period. Such a situation decreases the correlation between the flooding area and river



a) Biebrza. SAR: 2017-04-06, Optical: 2017-04-05.



b) Sacramento. SAR: 2023-03-19 Optical: 2023-03-20.



c) Ouse SAR & Optical: 2015-12-29.

Fig. 3. Floodsar flood maps from 2D and 1D algorithms, reference flood map calculated using MNDWI thresholding, and the color-infrared (CIR, R = near-infrared, G = red, B = green) composition of Sentinel-2 bands in each case study (a-c). In the flood maps white pixels indicate flood (open water) and black pixels indicate not flood. Images were selected for the reference flood map dates, as indicated below the panels and in Fig. 1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

gauge observation. Yet, the relation between the flooding extent and discharge was identified by Floodsar, because the flooding extent was considerably higher during the two flood events than in other periods.

Both Floodsar algorithms produced false positives. Less false positives were produced when VH data was used and more when the algorithm relied only on the VV data. The false positives were due to low backscatter for both flat objects and open water. The solution to this problem is usually to perform post-processing, e.g. with the Height Above Nearest Drainage (HAND) method [32], which masks areas where flooding is unlikely.

To account for the nonlinear character of the relationship between the flooded area and river gauge observations, given the topography of the river valley, we initially tested the Spearman's rank correlation coefficient along with Pearson's correlation. However, unlike Pearson's correlation, these tests showed that the Spearman's correlation is maximal for very small t_0 threshold values, leading to a vast underestimation of the flood area. The Pearson's correlation approach proved superior, likely because under steady-state conditions for a river valley with a trapezoidal-like cross-section, the relationship between water level and flood area is linear.

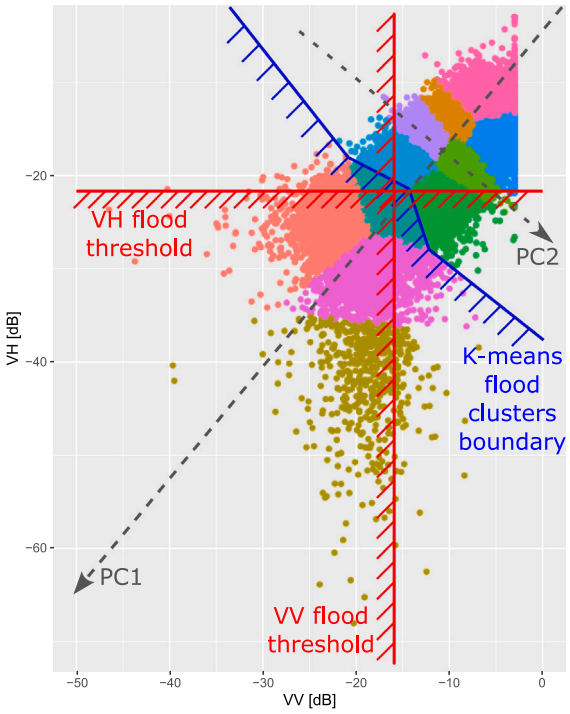


Fig. 4. The VV and VH backscatter data (points) for the Biebrza case study (37 SAR images 2017-03-01 to 2017-06-26) with annotated flood/no flood boundaries computed by the 1D/VV and 1D/VH algorithm (red lines) and by the 2D algorithm (blue lines). The side indicated by the diagonal sections on the red and blue boundary lines points on the "flooded" data points. The point color indicates membership to the k-means clusters in the 2D algorithm. Data is also annotated by the two principal component (PC) axes (dashed lines). Only randomly sampled 0.2% of data points are shown for clarity. The data were clipped to −3 dB before clustering. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Both Floodsar algorithms are influenced by the study site's AOI. A proper selection of the AOI is essential for Floodsar to identify thresholds or clusters. In general, it is required that the AOI is focused on the floodplain. If the flooding area is considerably smaller than the AOI area there is a possibility that the area of false positive pixels will be greater than the area of flood pixel. In such a case a relation between river gauge observations and the flooding area will not be identified.

4. Impact

Floodsar effectively captures the flooding extent during high water levels, close to the flood peak, but also during low water level conditions. This makes Floodsar suitable for investigating full flood dynamics, not only the high water level events. Histogram separation techniques, such as [25], which are often used for SAR flood mapping, yield good results when a flood covers a considerable fraction of an image because the backscatter histogram is bimodal. In Floodsar, the flood threshold (1D), or the flood clusters (2D) are determined based on ancillary physical observations, not the histogram. As a result, during low water levels, Floodsar gives much more realistic results than e.g. Otsu [25] method and therefore, may be used to replace algorithms based on histogram separation or be used as a member of an ensemble prediction.

We see the applicability of Floodsar in two types of applications, which require investigation of dynamic flood behavior, from low water levels to a flood peak. First, Floodsar can be implemented in real-time monitoring. In this case, Floodsar should be integrated with early access data storage, such as provided by the ASF and ESA. Optimally, an instance of Floodsar should run on a cloud that has direct access to near real-time Sentinel-1 data. An example of a platform with computation and storage capabilities is the Amazon Web Service (AWS), which currently operates the Sentinel-1 data archive used by the ASF OpenSarLab. Floodsar recognizes filenames processed by the ASF Hyp3, therefore integration with this cloud is straightforward.

The second type of application in which Floodsar can be used is processing long time series of SAR data for research or analytic purposes especially when flood extent maps are not available. An example of such an application could be producing flooding frequency maps as predictors for vegetation modeling [33,34]. In [1,2] a Floodsar application is presented where a long-time-series of flood maps was used for hydrological model validation. Further, the hydrological model output was used to predict vegetation development for two centuries of past and future climate [35].

A similar type of application in which Floodsar can be used is to generate flood extents for assimilation or validation in hydrological or hydrodynamic models. These are important fields because if assimilation is properly scheduled in the forecast time it can improve the model projections [36]. Also, using flood extent maps for model validation leads to optimal model selection [37].

Yet another type of application related to the former aspect is the hydrological model calibration. This field is particularly important because it leads to more reliable models, which can offer better flood susceptibility analysis [38,39]. Yet, Floodsar has not been tested in this field and currently, if a small number of flood maps are enough for a calibration problem, a cloud-free MNDWI thresholding offers a reasonable solution.

The case studies have shown that the 2D algorithm results overall in better agreement of the identified flood extent with the reference maps than in the 1D algorithm. Also, the 2D algorithm is more universal by providing valid results in all case studies. Yet in most cases the difference between 2D and 1D/VV algorithms is negligible. Given the greater complexity and longer computation times in the 2D algorithm, a user should test the performance of each algorithm in a study site before using Floodsar in a production instance.

Finally, Floodsar software offers an example of a physically informed machine learning model [40], which has attracted more and more attention recently. Although Floodsar uses relatively basic algorithms (clustering, thresholding, optimization), it shows that time-variable phenomena can be identified on a series of images given a summarizing variable (river gauge observations in our case) and physical constraints (water extent increases with increasing river gauge observations). Therefore, the Floodsar algorithm can potentially be used for identifying other time-variable problems, such as land-cover change given e.g. population data.

5. Conclusions

In this study, we presented the details of Floodsar software and demonstrated its performance is three case studies. Floodsar features two algorithms for automatic flood detection on SAR dual-pol VV+VH imagery. Except for the time series of SAR imagery, Floodsar requires also corresponding river gauge observations, such as water levels or discharges, to correlate the flooding extent to a physical phenomenon. The first algorithm (1D) uses one SAR polarization at a time. The 1D algorithm identifies the flooded/not flooded threshold based on the correlation of the flooded area with the river gauge observation(s). The second algorithm (2D), which was introduced here, uses two polarizations at the same time. The 2D algorithm uses the same principle but performs clustering of the imagery to choose the total number of clusters out of which the flood clusters are indicated.

The main limitation of methods used for flood mapping in Floodsar is that they are not capable of identifying flooding extent obscured by vegetation, in the radar shadow of other objects, or when a double-bouncing effect is present such as flooded buildings. Another limitation concerns false positive detection in arid regions or flat surfaces. In

either case, post-processing techniques exist that can be implemented but are out of the scope of the current version of Floodsar.

Floodsar is an example of physically informed machine learning. Therefore we plan to include more complex machine learning algorithms in Floodsar, such as deep neural networks, capable of capturing spatial and temporal patterns from the imagery to detect floods.

CRediT authorship contribution statement

Tomasz Berezowski: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Software, Validation, Writing – original draft, Writing – review & editing, Visualization. **Szymon Niemiec:** Software, Visualization, Writing – original draft. **Andrzej Chybicki:** Software, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data to run the Sacramento case study in Floodsar are available in a repository:

Input files for the Floodsar software (Original data) (Bridge of Knowledge).

Acknowledgments

This research was partially financed by a grant: 2017/26/D/ST10/00665 funded by the National Science Centre, Poland. Computations were carried out using the computers of Centre of Informatics Tricity Academic Supercomputer & Network in Poland.

Data for the Sacramento case experiments are available in [41].

We thank two anonymous reviewers whose comments contributed to significant improvement of this study.

References

[1] Berezowski T, Bielinski T, Osowicki J. Flooding extent mapping for synthetic aperture radar time series using river gauge observations. IEEE J Sel Top Appl Earth Observ Remote Sens 2020;13:2626–38. http://dx.doi.org/10.1109/jstars.2020.2995888.

[2] Berezowski T, Partington D. Impact of climate change on water sources and river-floodplain mixing in the natural wetland floodplain of Biebrza River. Water Resour Res 2023;59:e2023WR035836. http://dx.doi.org/10.1029/2023WR035836.

[3] Xu H. Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery. Int J Remote Sens 2006;27(14):3025–33. http://dx.doi.org/10.1080/01431160600589179.

[4] Singh KV, Setia R, Sahoo S, Prasad A, Pateriya B. Evaluation of NDWI and MNDWI for assessment of waterlogging by integrating digital elevation model and groundwater level. Geocarto Int 2014;30(6):650–61. http://dx.doi.org/10.1080/10106049.2014.965757.

[5] Manavalan R. SAR image analysis techniques for flood area mapping - Literature survey. Earth Sci Inform 2016;10(1):1–14. http://dx.doi.org/10.1007/s12145-016-0274-2.

[6] Martinis S, Kersten J, Twele A. A fully automated TerraSAR-X based flood service. ISPRS J Photogramm Remote Sens 2015;104:203–12. http://dx.doi.org/10.1016/j.isprsjprs.2014.07.014.

[7] Landuyt L, Van Wesemael A, Schumann GJ-P, Hostache R, Verhoest NEC, Van Coillie FMB. Flood mapping based on synthetic aperture radar: An assessment of established approaches. IEEE Trans Geosci Remote Sens 2019;57(2):722–39. http://dx.doi.org/10.1109/tgrs.2018.2860054.

[8] Nico G, Pappalepore M, Pasquariello G, Refice A, Samarelli S. Comparison of SAR amplitude vs. coherence flood detection methods - A GIS application. Int J Remote Sens 2000;21(8):1619–31. http://dx.doi.org/10.1080/014311600209931.

[9] Bazi Y, Bruzzone L, Melgani F. An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images. IEEE Trans Geosci Remote Sens 2005;43(4):874–87. http://dx.doi.org/10.1109/tgrs.2004.842441.

[10] Townsend P. Mapping seasonal flooding in forested wetlands using multi-temporal radarsat SAR. Photogram Eng Remote Sens 2001;67:857–64.

[11] Wang Y. Seasonal change in the extent of inundation on floodplains detected by JERS-1 synthetic aperture radar data. Int J Remote Sens 2004;25(13):2497–508. http://dx.doi.org/10.1080/01431160310001619562.

[12] Melancon AM, Molthan AL, Griffin RE, Mecikalski JR, Schultz LA, Bell JR. Random forest classification of inundation following hurricane florence (2018) via L-band synthetic aperture radar and ancillary datasets. Remote Sens 2021;13(24):5098. http://dx.doi.org/10.3390/rs13245098.

[13] Tavus B, Can R, Kocaman S. A CNN-based flood mapping approach using sentinel-1 data. ISPRS Ann Photogramm, Remote Sens Spat Inf Sci 2022;3:549–56. http://dx.doi.org/10.5194/isprs-annals-v-3-2022-549-2022.

[14] Amitrano D, Di Martino G, Iodice A, Riccio D, Ruello G. Unsupervised rapid flood mapping using sentinel-1 GRD SAR images. IEEE Trans Geosci Remote Sens 2018;56(6):3290–9. http://dx.doi.org/10.1109/tgrs.2018.2797536.

[15] Panahi M, Rahmati O, Kalantari Z, Darabi H, Rezaie F, Moghaddam DD, et al. Large-scale dynamic flood monitoring in an arid-zone floodplain using SAR data and hybrid machine-learning models. J Hydrol 2022;611:128001. http://dx.doi.org/10.1016/j.jhydrol.2022.128001.

[16] Cloud to Street. The global flood database website. 2023, Online URL https://global-flood-database.cloudtostreet.ai/#interactive-map. [Accessed 16 November 2023].

[17] Tellman B, Sullivan JA, Kuhn C, Kettner AJ, Doyle CS, Brakenridge GR, et al. Satellite imaging reveals increased proportion of population exposed to floods. Nature 2021;596(7870):80–6. http://dx.doi.org/10.1038/s41586-021-03695-w.

[18] NOAA. NOAA satellite proving ground global flood website. 2023, Online URL https://realearth.ssec.wisc.edu/?products=RIVER-FLDglobal-composite1.-100. [Accessed 16 November 2023].

[19] Sjoberg B, Li S, Sun D. Global flood mapping services from JPSS. In: IGARSS 2018 - 2018 IEEE international geoscience and remote sensing symposium. IEEE; 2018, http://dx.doi.org/10.1109/igarss.2018.8517357.

[20] Li S, Sun D, Goldberg MD, Sjoberg B, Santek D, Hoffman JP, et al. Automatic near real-time flood detection using Suomi-NPP/VIIRS data. Remote Sens Environ 2018;204:672–89. http://dx.doi.org/10.1016/j.rse.2017.09.032.

[21] Wasdi. Wasdi website. 2023, Online URL https://www.wasdi.cloud. [Accessed 16 November 2023].

[22] Chini M, Hostache R, Giustarini L, Matgen P. A hierarchical split-based approach for parametric thresholding of SAR images: Flood inundation as a test case. IEEE Trans. Geosci. Remote Sens. 2017;55(12):6975–88. http://dx.doi.org/10.1109/tgrs.2017.2737664.

[23] Torres R, Snoeij P, Geudtner D, Bibby D, Davidson M, Attema E, et al. GMES sentinel-1 mission. Remote Sens Environ 2012;120:9–24. http://dx.doi.org/10.1016/j.rse.2011.05.028.

[24] Hogenson K, Kristenson H, Kennedy J, Johnston A, Rine J, Logan T, et al. Hybrid pluggable processing pipeline (HyP3): A cloud-native infrastructure for generic processing of SAR data. 2023, http://dx.doi.org/10.5281/ZENODO.4646138.

[25] Otsu N. A threshold selection method from gray-level histograms. IEEE Trans. Syst., Man, Cybern. 1979;9(1):62–6. http://dx.doi.org/10.1109/tsmc.1979.4310076.

[26] Lloyd S. Least squares quantization in PCM. IEEE Trans. Inf. Theory 1982;28(2):129–37. http://dx.doi.org/10.1109/tit.1982.1056489.

[27] Alaska Satellite Facility. Sentinel-1 RTC product guide. 2023, Online URL https://hyp3-docs.asf.alaska.edu/guides/rtc_product_guide/. [Accessed 14 June 2023].

[28] USGS. United States geological survey station 11455420. 2023, Online URL https://waterdata.usgs.gov/nwis/uv/?referred_module=sw. [Accessed 14 June 2023].

[29] NRFA. UK centre for ecology & hydrology, national river flow archive, station 27009. 2023, Online URL https://nrfa.ceh.ac.uk/data/search. [Accessed 14 June 2023].

[30] IMGW-PIB. Institute for meteorology and water management - National Research Institute (IMGW-PIB), station 153220100. 2023, Online URL https://danepubliczne.imgw.pl/data/dane_pomiarowo_obserwacyjne/dane_hydrologiczne/dobowe/. [Accessed 14 June 2023].

[31] Clement M, Kilsby C, Moore P. Multi-temporal synthetic aperture radar flood mapping using change detection. J Flood Risk Manag 2017;11(2):152–68. http://dx.doi.org/10.1111/jfr3.12303.

[32] Rennó CD, Nobre AD, Cuartas LA, Soares JV, Hodnett MG, Tomasella J, et al. HAND, a new terrain descriptor using SRTM-DEM: Mapping terra-firme rainforest environments in Amazonia. Remote Sens Environ 2008;112(9):3469–81. http://dx.doi.org/10.1016/j.rse.2008.03.018.

[33] Murray-Hudson M, Wolski P, Cassidy L, Brown MT, Thito K, Kashe K, et al. Remote sensing-derived hydroperiod as a predictor of floodplain vegetation composition. Wetl Ecol Manag 2014;23(4):603–16. http://dx.doi.org/10.1007/s11273-014-9340-z.

[34] Tan J, Chen M, Ao C, Zhao G, Lei G, Tang Y, et al. Inducing flooding index for vegetation mapping in water-land ecotone with Sentinel-1 & Sentinel-2 images: A case study in Dongting Lake, China. Ecol Indic 2022;144:109448. http://dx.doi.org/10.1016/j.ecolind.2022.109448.

[35] Berezowski T, Wassen M. Using water sources extent during inundation as a reliable predictor for vegetation zonation in a natural wetland floodplain. Ecol Indic 2023;154:110854. http://dx.doi.org/10.1016/j.ecolind.2023.110854.

[36] Hostache R, Chini M, Giustarini L, Neal J, Kavetski D, Wood M, et al. Near-real-time assimilation of SAR-derived flood maps for improving flood forecasts. Water Resour Res 2018;54(8):5516–35. http://dx.doi.org/10.1029/2017wr022205.

[37] Nguyen TH, Ricci S, Fatras C, Piacentini A, Delmotte A, Lavergne E, et al. Improvement of flood extent representation with remote sensing data and data assimilation. IEEE Trans Geosci Remote Sens 2022;60:1–22. http://dx.doi.org/10.1109/tgrs.2022.3147429.

[38] Domeneghetti A, Tarpanelli A, Brocca L, Barbetta S, Moramarco T, Castellarin A, et al. The use of remote sensing-derived water surface data for hydraulic model calibration. Remote Sens Environ 2014;149:130–41. http://dx.doi.org/10.1016/j.rse.2014.04.007.

[39] Gobeyn S, Wesemael AV, Neal J, Lievens H, Eerdenbrugh KV, Vleeschouwer ND, et al. Impact of the timing of a SAR image acquisition on the calibration of a flood inundation model. Adv Water Resour 2017;100:126–38. http://dx.doi.org/10.1016/j.advwatres.2016.12.005.

[40] Champion K, Zheng P, Aravkin AY, Brunton SL, Kutz JN. A unified sparse optimization framework to learn parsimonious physics-informed models from data. IEEE Access 2020;8:169259–71. http://dx.doi.org/10.1109/access.2020.3023625.

[41] Berezowski T. Input files for the floodsar software. 2023, http://dx.doi.org/10.34808/6NFS-6Q42.