Science of Remote Sensing 11 (2025) 100203

Contents lists available at ScienceDirect

Science of Remote Sensing

journal homepage: www.sciencedirect.com/journal/science-of-remote-sensing

DeepSARFlood: Rapid and automated SAR-based flood inundation mapping using vision transformer-based deep ensembles with uncertainty estimates

Nirdesh Kumar Sharma [a], Manabendra Saharia [a,b,*]

[a] Department of Civil Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India
[b] Yardi School of Artificial Intelligence, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India

ARTICLE INFO

Keywords:
Flood mapping
Synthetic aperture radar
Deep ensembles
Uncertainty estimation
Vision-transformer

ABSTRACT

Rapid and automated flood inundation mapping is critical for disaster management. While optical satellites provide valuable data on flood extent and impact, their real-time usage is limited by challenges such as cloud cover, limited vegetation penetration, and the inability to operate at night, making real-time flood assessments difficult. Synthetic Aperture Radar (SAR) satellites can overcome these limitations, allowing for high-resolution flood mapping. However, SAR data remains underutilized due to less availability of training data, and reliance on labor-intensive manual or semi-automated change detection methods. This study introduces a novel end-to-end methodology for generating SAR-based flood inundation maps, by training deep learning models on weak flood labels generated from concurrent optical imagery. These labels are used to train deep learning models based on Convolutional Neural Networks (CNN) and Vision Transformer (ViT) architectures, optimized through multitask learning and model soups. Additionally, we develop a novel gain algorithm to identify diverse ensemble members and estimate uncertainty through deep ensembles. Our results show that ViT-based and CNN-ViT hybrid architectures significantly outperform traditional CNN models, achieving a state-of-the-art Intersection over Union (IoU) score of 0.72 on the Sen1Floods11 test dataset, while also providing uncertainty quantification. These models have been integrated into an open-source and fully automated, Python-based tool called DeepSARFlood, and demonstrated for the Pakistan floods of 2022 and Assam (India) floods of 2020. With its high accuracy, processing speed, and ability to estimate uncertainty, DeepSARFlood is optimized for real-time deployment, processing a $1^\circ \times 1^\circ$ (12,100 km^2) area in under 40 s, and will complement upcoming SAR missions like NISAR and Sentinel 1-C for flood mapping.

1. Introduction

Floods are the most damaging natural hazards, causing economic losses in the order of billions of dollars. Recent analyses indicate that the total economic burden of flooding in the United States alone ranges between $179.8 billion and $496 billion each year ("Economic Cost of Flooding," 2024). Over the past few decades, floods have grown in magnitude and frequency due to increased anthropogenic activities and climate change (AghaKouchak et al., 2020; Hamidi et al., 2023; Singh and Dhanya, 2024). Consequently, the damage due to floods has significantly multiplied, driving attention to flood management strategies, particularly near real-time flood mapping and monitoring. Real-time high-resolution flood inundation mapping and monitoring are crucial to aid rescue and evacuation during floods, assessing property loss and damages, and facilitating damage recovery decisions (Benoudjit and Guida, 2019; Li and Demir, 2023; Shen et al., 2019b). However, the development of such maps cannot be done using optical satellite data, which is often obscured by cloud cover during flood events. Synthetic Aperture Radar (SAR) data can penetrate clouds and provide timely inundation information; however, interpreting SAR data is complex. In this study, we leverage significant advancements in artificial intelligence and satellite imagery to develop a fully automated framework for flood inundation mapping using SAR data and weak labels developed from optical data (see Fig. 7).

Traditionally flood inundation mapping has been done using ground surveys and numerical models. Ground surveys are expensive, and dangerous due to unfavorable weather conditions and the collapse of transportation and communication systems (Jiang et al., 2021; Manjusree et al., 2012). The numerical models, especially hydrologic and hydrodynamic models, have a limited utility in flood extent mapping

* Corresponding author. Department of Civil Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India.
E-mail address: msaharia@iitd.ac.in (M. Saharia).

https://doi.org/10.1016/j.srs.2025.100203
Received 28 September 2024; Received in revised form 18 January 2025; Accepted 4 February 2025
Available online 7 February 2025

and emergency response, primarily due to the large extent of ungauged basins, high costs of maintaining rain and stream gauges, and large computational expense (Klemas, 2015; Singh et al., 2020). Moreover, large river basins often extend beyond a country's political boundaries, preventing sharing of gauge data required for calibration (Gleason and Hamdan, 2017).

Recently, satellite imagery has emerged as a consistent source of information for monitoring flood extent especially in data scarce regions due to its vast coverage, low cost and time expenses, and independence from in-situ measurements. Optical satellite imagery uses the high absorptive capacity of water in the short-wave infrared spectrum (SWIR) and high reflectivity in the near-infrared (NIR) spectrum to map floods. Although previous research has demonstrated using optical data from Landsat, Sentinel-2, and MODIS for surface water and flood mapping (Goffi et al., 2020; Isikdogan et al., 2017); during floods, optical remote sensing data is inadequate as the clear view of the earth's surface is obstructed by clouds (DeVries et al., 2020; Yan et al., 2015). Synthetic Aperture Radar (SAR) data has been extensively used to map flood inundation due to its ability to penetrate the clouds and some vegetation, and day and night operation (Amitrano et al., 2018; Manavalan, 2017; Sharma et al., 2022). Due to the specular reflection of SAR waves by the water surface, minimal backscatter is returned to the sensor making the surface water appear dark and homogenous (Sherpa and Shirzaei, 2022). However, flood extent development from SAR data is not straightforward since the backscatter values also vary due to location, land use, meteorological conditions, and satellite system.

SAR based flood inundation traditionally has been done using thresholding and change detection methods. The threshold-based methods require a bimodal histogram of pixels or seed water pixels to find an optimal value below which a pixel can be classified as water. However, finding an optimal threshold consumes a lot of time, delaying the final production of flood maps (Shen et al., 2019a). Change detection approaches use a pre-flood and during-flood SAR backscatter to segment areas with unusually low backscatter response as flooded (Clement et al., 2018; Li et al., 2018). The change detection methods are sensitive to parameter differences in SAR images and are unable to identify slightly inundated regions (Jiang et al., 2021). Both methods are also limited since they can only be used on a single polarized radar image, thereby losing the information contained in other SAR polarization data. Also, both algorithms require manual parameter tuning, which adds subjectivity and presents a challenge for the automated mapping of complex and large floods (Jiang et al., 2021).

Recently artificial intelligence-based methods have become popular for flood inundation mapping due to their easy automation, high accuracy, and ability to integrate multiple auxiliary datasets. While traditional machine learning algorithms like random forest have been used to map floods from SAR data, they are limited by their pixel-based nature and inability to capture spatial information crucial for flood mapping (Feng et al., 2015; Woznicki et al., 2019). Deep learning models, especially Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs), present a new paradigm in flood mapping by not only learning complex relationships between SAR data and flood labels but also leveraging the spatial structure of data. Deep learning models are not only more accurate, but also group homogenous pixels into patches at the object level, making them more resistant to speckles than other flood segmentation methods (Shen et al., 2019b). CNN-based U-Net architectures, especially those with ResNet encoders, have been used frequently in flood mapping applications(Bereczky et al., 2022; Konapala et al., 2021; Li and Demir, 2023). Some studies have also used attention mechanisms as well as advanced architectures like U-Net++ and DeepLabV3 (Paul and Ganju, 2021). Recently, hybrid architectures combining U-Net and atrous convolutions inspired by DeepLab have demonstrated improved accuracy compared to standalone U-Net and DeepLab models (J. Wang et al., 2022a). However CNN based models have a limited receptive field which can lead to a decrease in accuracy while mapping floods (Saleh et al., 2024). Vision transformers(ViTs) can model long range dependencies and have been used in multiple tasks such as ship identification and change detection (Chen et al., 2022; Yasir et al., 2024). ViT based models have also been developed for flood change detection using Siamese networks (Saleh et al., 2024).

Despite all their benefits, both CNN and ViT based deep learning models require a large amount of training data and computational resources. In recent years, the advancements in Graphical Processing Unit (GPU) technology and parallelized computation has allowed deep learning models to be trained faster and deployed in real-time. However, the problem of developing labeled data for training deep learning models remains. Unlike optical imagery, where labeling is straightforward, SAR is a complex sensor, and the imagery is incomprehensible to human eyes thereby making developing direct labels difficult. Traditionally, high-quality labeled datasets for SAR-based flood mapping have been generated by matching SAR data with concurrent cloud-free optical imagery (Bonafilia et al., 2020). In this approach inundated surfaces are identified using the optical imagery through visual inspection and the application of water indices such as the Normalized Difference Water Index (NDWI) and the Modified Normalized Difference Water Index (MNDWI) (Bonafilia et al., 2020; Wieland et al., 2024). However, the manual labeling process is both labor-intensive and time-consuming, resulting in a limited number of hand-labeled datasets. This scarcity can lead to models trained insufficient data, which may produce biased predictions and fail to adequately capture the complexities inherent in flood mapping scenarios. To address these limitations, recent advancements have introduced models that utilize weak labels, significantly reducing the need for human intervention during the labeling process (Bonafilia et al., 2020; Garg et al., 2023).

While the transition to weak labeling techniques helps creation of larger training datasets more efficiently and enhances model generalization by incorporating diverse cases, it also presents multiple challenges. Specifically, low-quality weak labels can compromise the overall quality of the models (Garg et al., 2023). When the label data is weak, more than one set of model weights can produce similar results, and the model may optimize itself for specific conditions. This problem can be overcome by combining the outputs of multiple deep learning models, also known as deep ensembles. Empirical evidence suggests that deep ensembles are a promising approach for enhancing the accuracy, uncertainty quantification, and out-of-distribution robustness of deep learning models (Fort et al., 2020). However, identifying ensemble members is a complex task since they must be diverse, and learn different features, thereby making their combination more accurate and robust.

Uncertainty estimation is another significant aspect of flood mapping and can be attributed to noisy and incomplete data or model design. Uncertainty quantification is necessary for real-time flood management to avoid making inaccurate decisions, and for integrating satellite data into numerical models using data assimilation filters (Giustarini et al., 2016). Uncertainty estimation for deep learning-based flood inundation models has been an active area of research and has been developed using Bayesian approaches and Monte Carlo dropout networks (Dechesne et al., 2021; Hertel et al., 2023). Although Bayesian neural networks are theoretically robust, they do not perform as well as deep ensembles in practice, particularly under dataset shift (Fort et al., 2020).

This study aims to overcome existing limitations in flood mapping by developing an automated framework for generating weak labels using concurrent optical and SAR data. These labels are used to train multiple ViT and CNN based models for rapid and automated flood mapping. Further, we present an improved optimization scheme involving multi-task learning and model soups to improve individual deep learning models. Additionally, a novel gain algorithm is used to identify diverse ensemble members and develop uncertainty estimates. Finally for operational deployment, we present DeepSARFlood, a modular and highly optimized Python tool tailored for real-time usage. DeepSARFlood is designed to be flexible, allowing users to adapt the framework according to operational needs while ensuring efficient processing

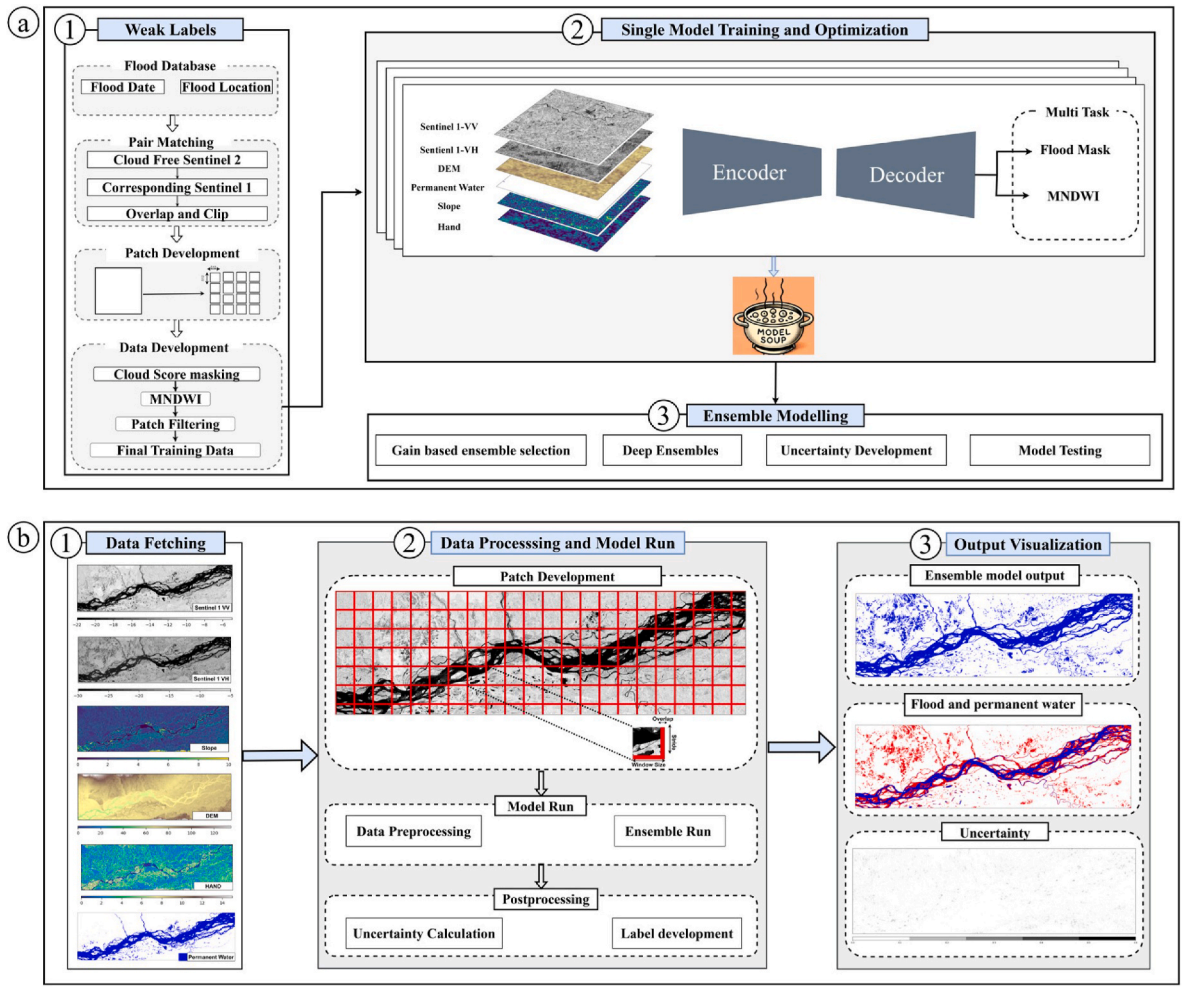

Fig. 1. a) Shows the End-to-end methodology for training deep learning models using weak labels 1) Shows the process of development of weak labels using Sentinel 1 and Sentinel 2 data 2) shows development of optimized individual models using multitask learning and model soups c) shows the ensemble modelling and model validation process.
b) Shows how the ensemble models are deployed in DeepSARFlood 1) Shows the automated data fetched process from GEE 2) Shows the process of patch development and model run 3) Shows the output visualization module of DeepSARFlood.

of SAR data for flood mapping applications. Using these advancements, this study seeks to improve the accuracy and efficiency of flood mapping processes, ultimately contributing to more effective flood management strategies.

2. Data sources

The following datasets were utilized in this study.

2.1. Sentinel-1 (S1)

S1 is an active microwave remote sensing satellite operating at a frequency of 5.405 GHz in a dual-polarization mode and has a revisit time of 12 days (Torres et al., 2012). Due to its high spatial resolution and capability to penetrate clouds and shallow vegetation, S1 images are extensively employed as a source for timely and cost-efficient flood inundation mapping by governments and relief organizations (Twele et al., 2016). We use S1 Ground Range Detected (GRD) scenes which are processed to develop backscatter coefficient (σ°) (https://developers.google.com/earth-engine/guides/sentinel1). The σ° value measures how the terrain scatters the radiation and depends on the terrain characteristics, geometry of the terrain, and the interaction of the terrain with ground elements. When the terrain gets flooded, the water layer reflects the SAR radiation in a specular manner leading to a very low backscatter, hence, water appears dark in SAR images. This allows us to separate water from other features.

2.2. Sentinel-2 (S2)

S2 is a wide swath satellite that provides multispectral images at high resolution for continuous earth surface monitoring. S2 has 13 bands out of which we used the short-wave infrared and visible green bands to calculate a normalized difference, MNDWI for water mapping (Xu, 2006). MNDWI has been extensively used by studies to develop water extent layers (Nagel et al., 2023; Singh et al., 2015). Despite its high accuracy in mapping water bodies, S2 faces significant constraints in flood mapping due to frequent cloud cover, which can obscure the imagery during crucial flooding events. In this study we use S2 data to develop label data for S1 by using concurrent S1-S2 imagery.

2.3. Digital elevation model (DEM)

A DEM is a digital representation of the topography or elevation of a terrain, typically generated from satellite or Unmanned Aerial Vehicles (UAVs). Each pixel in a DEM contains a numeric value representing the elevation of that location on the earth's surface. DEMs are especially useful in identification of low-lying flood plains (Samela et al., 2016). In this study we use NASA DEM, which was developed by reprocessing the

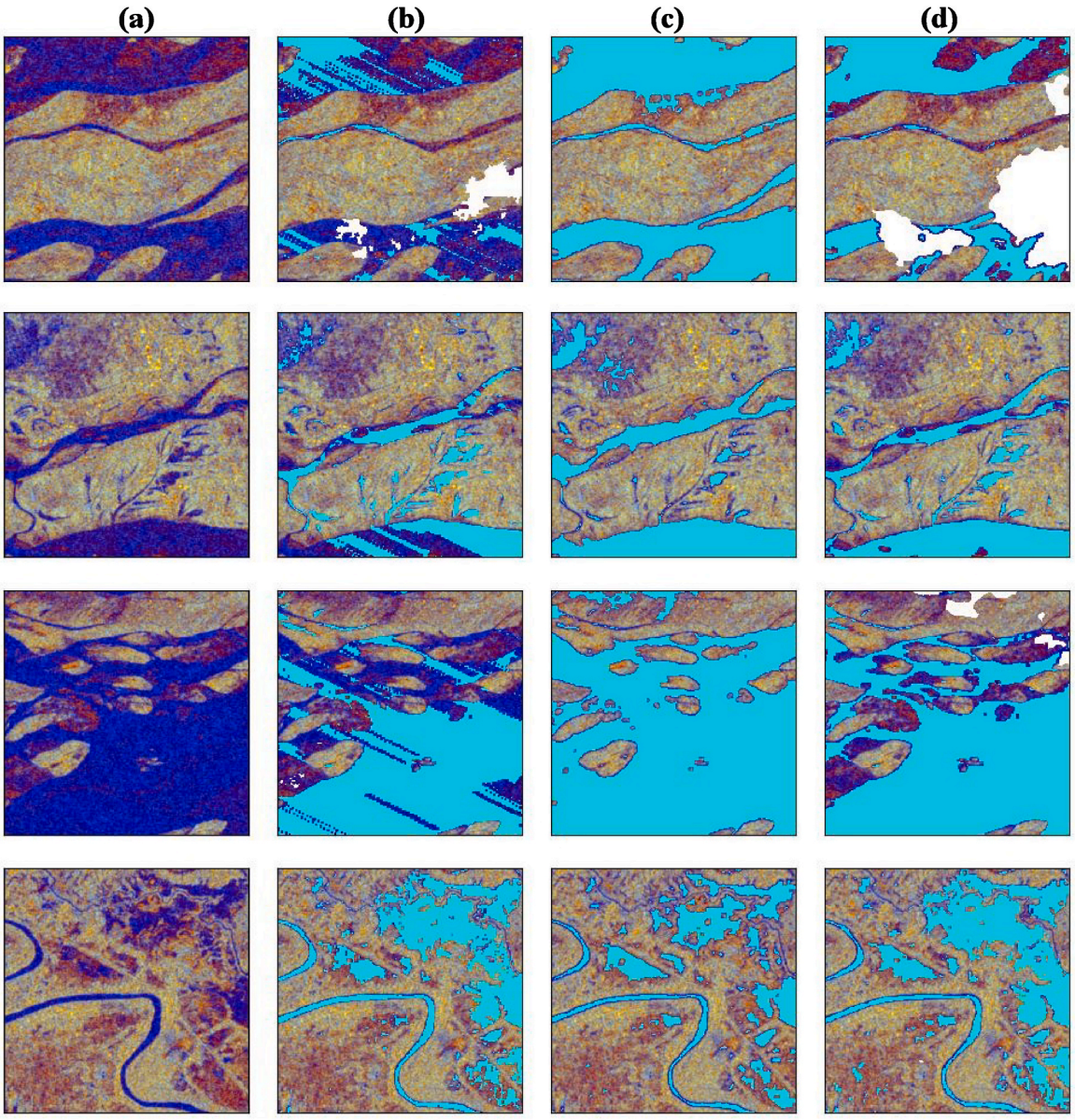

Fig. 2. Comparison of Sen1Floods11 weak labels with weak labels developed by our methodology. a) S1 patch b) Sen1Floods11 flood label c) Sen1Floods11 OTSU label d) label developed by our methodology. Note: The a, b, c images are from Sen1Floods11 data and the basemap is a visualization of S1.

SRTM data using ancillary datasets and is void-free unlike its predecessor. NASA DEM is one of the most accurate DEMs amongst open source global DEMs (Uuemaa et al., 2020).

2.4. Slope

Slope refers to the steepness of the terrain and it impacts the drainage pattern of an area by dictating the water flow across the landscape. For a gentle slope the runoff gets stored and disperses slowly over time causing floods, conversely on a steep slope the water flow is rapid due to gravitational forces making them non-susceptible to floods. We derive the slope from NASA DEM using Google Earth Engine (GEE) (Gorelick et al., 2017). We use the slope as an input as well as a postprocessor since no water can accumulate at a slope greater than 5% (Tazmul Islam and Meng, 2022).

2.5. Permanent water

Since permanent water and flood water have different signatures including permanent water data adds to the robustness of the model predictions (Li and Demir, 2023). We use the JRC permanent water layer that provides information on the location and extent of surface water bodies based on Landsat data at a spatial resolution of 30 m (Pekel et al., 2016).

2.6. Hand

Height Above Nearest Drainage (HAND) is a terrain model that normalizes topography to the relative heights along the drainage network. It quantifies the local drainage potential such that each pixel represents the vertical distance in meters to the nearest drainage (Nobre et al., 2011). Consequently, it can be used to identify regions that are more prone to flood inundation. We use a global HAND dataset developed using SRTM DEM (Donchyts et al., 2016).

3. Methodology

The aim of this study is to develop a supervised deep learning

Algorithm 1: Ensemble member selection using gain algorithm

Input : Models (Model 1, ..., Model m),
Images (Image1, ..., Image n),
Image Labels (Label(image 1),...,Label(image n))
Threshold value τ and
Maximum % information change p

Output: Gain Value for ensemble members

```
/* Get Maximum accuracy per image and maximum accuracy model */
(IOU)_max ← []
(IOU)_model ← []
foreach i ∈ Images do
    Accuracy ← 0
    foreach j ∈ Models do
        temp=IOU(Predict(j, i),Label(i))
        if (temp > Accuracy) then
            Accuracy = temp
            Model =j
    (IOU)_max.append(Accuracy)
    (IOU)_model.append(Model)
/* Weak model development */
D_s^weak ← []
foreach i ∈ [1, ...., n] do
    if (IOU)_max [i] < τ) then
        D_s^weak . append(Image[i] ∪ AUG (Image[i]))
Model(m+1)← Train (D_s^weak)
/* (IOU)_base & (IOU)_model updation with weak model */
foreach i ∈ [1, ...., n] do
    Accuracy = (IOU)_max[i]
    temp = IOU(Predict(model(m+1), image[i]),Label(image[i]))
    if (temp > Accuracy) then
        (IOU)_max[i] = temp
        (IOU)_model [i]= Model(m+1)
/* Update images after removing true weak images */
index← []
foreach i ∈ [1, ...., n] do
    if ((IOU)_max [i] > τ) then
        index.append[i]
Images=Images[index]
Labels=Labels[index]
(IOU)_max=(IOU)_max[index]
(IOU)_model=(IOU)_model[index]
n=len(index)
/* Base model selection and One hot encoding */
Model_base ← Mode (IOU)_model
(IOU)_OHE ← OHE ((IOU)_model)
/* Residue calculation and gain formulation */
Gain_value←[ ]
Gain_model←[ ]
foreach j ∈ Models do
    (IOU)_j←[ ]
    foreach i ∈ Images do
        (IOU)_j.append(IOU(Predict(j,i),Label(i))
    Residue = (IOU)_max - max((IOU)_base,(IOU)_Model1...,(IOU)_j )
    Product←[0]_m*n
    foreach i ∈ [1, ...., n] do
        Product[i]= (IOU)_OHE [i] * Residue [i]
    Gain_list←[ ]
    foreach i ∈ [1, ...., m] do
        Gain_list.append(product[:,i].sum())
    Gain_value.append( max(Gain_list))
    Gain_model.append(argmax(Gain_list))
/* Information change calculation and ensemble model selection */
Ensemble_model←[ ]
%Information=(Gain_value*100)/Sum(Gain_value)
foreach i ∈ [1, ...., m] do
    if (( %Information[i] - %Information[i + 1])> p) then
        Ensemble_model.append(Gain_model[i])
return Gain_value,Gain_model,Ensemble_model
```

Fig. 3. Pseudocode of gain algorithm.

semantic segmentation model for flood inundation with limited labeled data. We present an end-to-end framework to develop artificial intelligence based models for near real-time flood mapping using SAR data. The overall methodology is explained in Fig. 1.

3.1. Weak label development

A deep learning model trained on large and diverse datasets is less likely to encounter out of distribution scenarios during inference. Among the available training datasets for Synthetic Aperture Radar (SAR)-based flood mapping, such as NASA-IMPACT, S1S2-Water, and Sen1Floods11, the Sen1Floods11 dataset is the most widely utilized. Sen1Floods11 is a georeferenced dataset specifically designed for training and testing deep learning algorithms for flood mapping using Sentinel-1 imagery. It includes 446 hand mapped images and 4385 weak labeled images covering an extensive area across multiple biomes and ecoregions globally (Bonafilia et al., 2020). The dataset comprises both hand-labeled and weakly labeled data, enabling models to learn from a variety of examples. The areas which were difficult to identify as floods due to the presence of clouds and other artifacts have been masked in Sen1Floods11 (Bonafilia et al., 2020).

Although Sen1Floods11 has been used extensively for SAR based flood mapping (Bai et al., 2021; Konapala et al., 2021; Muszynski et al., 2022), it suffers from a few limitations. The Sen1Floods11 dataset contains a relatively small amount of high-quality, hand-labeled data, which is insufficient for developing general-purpose deep learning models. The weak labels associated with Sen1Floods11 are known to exhibit significant inaccuracies as shown in Fig. 2. Previous studies have attempted to address this issue by incorporating permanent water labels into the weak data (Garg et al., 2023). However, this approach only rectifies errors stemming from the absence of permanent water information, leaving other limitations, such as incorrect labels, unaddressed. Furthermore, models trained solely on the weak data from Sen1Floods11 tend to learn artifacts from these weak labels rather than capturing general patterns (Garg et al., 2023). These limitations underscore the need for improved labeling techniques prior to training deep learning models. To enhance the quality of the training data, this study proposes the development of high-quality weak labels that can provide a more reliable data for model training. We leverage recent advancements in cloud and cloud shadow masking. This simple and effective method for developing weak flood labels is implemented as a standalone GEE application, thereby enabling automation at a large scale. The process of developing weak labels is explained below.

a. The application requires the user to input flood dates and the area of interest. The algorithm returns the dates of all S2 images with a cloud fraction of less than 30%.
b. Using the S2 date information, the algorithm then identifies all the S1 images which are coincidental or within 1 day of S2 imagery. These are known as matched pairs.
c. The algorithm then calculates the geometry of the overlap between matched pairs and clips the S2 imagery to the overlap geometry. The clipped S2 image is then further split into non overlapping patches of size 512*512.
d. For all patches, we mask clouds and cloud shadows using google cloud score plus (Pasquarella et al., 2023) value of less than 0.6. We then create a water label using the Modified Normalized Difference Water Index (MNDWI) using equations (1) and (2)

$$\mathrm{MNDWI} = \frac{\mathrm{B11} - \mathrm{B3}}{\mathrm{B11} + \mathrm{B3}} \tag{1}$$

$$\text{if MNDWI} > 0.3 \text{ Label} = 1 \text{ else Label} = 0 \tag{2}$$

e. Further all patches where masked values are less than 40% and flooded values are more than 1% are selected for training.
f. For the selected patches we fetch corresponding S1 data and auxiliary data to be used as inputs of our model.

Our labeling methodology offers the following benefits over previous methods.

a. Full automation and parallelization: The proposed application is developed in GEE and is fully automated and parallelized and allows for optimized training data generation.
b. Improved cloud and cloud shadow masking: Masking clouds and cloud shadows is essential before developing water labels since cloud shadows have spectral signature similar to surface water (Li et al., 2013; Wang et al., 2019). We use google cloud score plus which is a state-of-the-art cloud and cloud shadows removal algorithm and is more sensitive to haze and cloud shadows (Pasquarella et al., 2023). Google cloud score plus provides a significantly improved alternative to cloud removal techniques.
c. Optimum label selection: Since flooding is a highly biased problem, we only select patches which have at least 1% flooded area and low cloud and cloud shadows (less than 40%). This ensures that the training data is of high quality and the computational power is not wasted on masked images which won't contribute to back-propagation of the loss.
d. Lag time: In contrast to existing studies that typically adopt a lag of 2 days between the capture S1 and S2 data capture our methodology employs a lag time of only 1 day. This formulation ensures that any potential changes in flood inundation between the acquisition of S1 and S2 images are minimized.

Using our proposed methodology, we develop weak labeled training dataset in contrast to Sen1Floods11 weak labeled data. Fig. 2 shows the comparison of samples of Sen1Floods11 weak data with the data labeled by our algorithm. In contrast to Sen1Floods11 weak labeled data, the weak labels developed by the proposed methodology efficiently capture the continuity of water pixels and mask pixels effectively.

3.2. Data split

In this study, we developed three distinct datasets to ensure comprehensive coverage of the problem space and to facilitate the evaluation of our proposed methods. An overview of the development and size of datasets is explained below.

3.2.1. Training data

The training data comprises of a curated dataset of global flood events and those listed in Sen1Flood11 weak labels. The metadata (location and date) obtained from this combination was used to develop labels using the algorithm proposed in section 3.1. This resulted in 1120 high quality patches (of the 4385 total patches) which conform to the criterion discussed in section 3.1 from Sen1Floods11. Additionally, of the 15341 patches identified from the global flood events, 3321 were incorporated in the final model development after visual validation. Thus, the total training dataset consists of 4441 image patches.

3.2.2. Validation data

The purpose of the validation data is to assess the model during training and prevent overfitting. Due to the weak labeling of the training data, an independent split for validation could potentially lead the model to learn artifacts specific to S2 data. To address this issue, we combined the training, and validation sets from the Sen1Floods11 hand-labeled dataset to create a validation set comprising of 341 images, ensuring a robust evaluation of the model.

3.2.3. Testing data

For testing, we utilized the 90 test images recommended by the Sen1Floods11 dataset. These images are manually labeled and have been employed in numerous prior studies making them ideal for comparative analysis against similar studies.

3.3. Model architecture

Our models use the standard encoder-decoder architecture commonly employed for semantic segmentation. The encoder extracts contextual features from input data and is rich in semantic information but poor in spatial information. The decoder up-samples the semantic information and combines it with spatial information from encoder layers, thereby allowing the network to combine information from multiple levels of abstraction. This formulation helps the network to produce accurate pixel-wise labels. The encoder mostly is a network which has excellent performance in image classification tasks and thereby have powerful feature extraction capabilities (He et al., 2022; L. Wang et al., 2022). The existing encoder architectures can be classified into Convolutional Neural Network based architectures (CNNs), Vision Transformer based architectures (ViTs) and CNN-ViT hybrid architectures all of which are explored in detail in this study. The CNN based encoders stack multiple convolutional and pooling layers to capture features at different scales. Although CNNs are highly effective at capturing local information, they struggle to capture global context due to their limited receptive fields. ViTs are inspired from language transformers and split the images into patches which are then regarded similar to sequences of words. ViTs use self-attention mechanisms that allows them to capture long-range dependencies and global context across the entire image. However, due to quadratic complexity of the self-attention mechanism ViTs require large computational power during training as well as inference. Recently computationally efficient variants of ViTs like Swin transformers have become popular. Swin transformer constructs a hierarchical representation like CNNs by beginning with small non-overlapping image patches and progressively merging neighbouring patches in deeper transformer layers. This approach allows the model to capture long-range dependencies. Also, since the number of patches in each window is fixed the computation complexity of Swin transformer scales linearly to image size unlike ViT transformers. Hybrid encoders like Multi-Axis Vision Transformer (MaxViT) combine the strengths of both CNNs and ViTs for efficient computation and scalability. MaxViT stacks convolution blocks with global and local attention blocks. This design helps in capturing enhanced local features and translation invariance using convolution. Meanwhile, the transformers capture global relationships making the feature extraction more robust. We explored and compared different CNN, ViT, and hybrid encoders.

The first decoder network was U-Net originally proposed for biomedical image segmentation (Ronneberger et al., 2015). The U-Net has a symmetric structure where each block in the decoder combines information from corresponding encoder using skip connections as well upsampled decoder. U-Net has since been used for multiple applications in natural hazards such as flood mapping(Konapala et al., 2021), landslide mapping and droughts. Recent advancements such as U-Net++ (Zhou et al., 2020), SegNet (Badrinarayanan et al., 2017), and DeepLabV3+ have overcome many limitations of the U-Net. For instance, U-Net++ used dense skip connections at different levels to overcome the requirement of training multiple models. SegNet provides more efficient and faster architecture at the cost of accuracy. On the other hand, DeepLabV3+ uses atrous convolution in the encoder features to learn features at different scales without degrading the spatial resolutions. For the decoder component, we investigate multiple architectures including U-Net, U-Net++, and DeepLabV3+.

To set up the encoder of our model we use Pytorch image models (Timm) library (Wightman et al., 2021, 2023). Although Timm is primarily designed for classification, it offers several advantages which can be used for developing segmentation models. Firstly, it contains high level implementations of pretrained state of the art classification models which can be used directly as encoders. Secondly, Timm offers intermediate feature extraction at multiple depths essential for developing skip-connections for decoders. Finally, Timm allows for the development of models with an arbitrary number of input channels, which is useful for remote sensing tasks.

3.4. Loss functions

A loss function measures how well the model's predictions match the true values. Since the training data is highly biased towards non-flood areas, using a regular loss function can lead to underestimation of flooded areas. Therefore, we employ multiple loss functions specifically designed for handling data imbalance and can identify flooded areas when most areas are non-flooded (Bischke et al., 2018). We use Dice loss, Jaccard loss, Tversky loss, and Focal Tversky loss which are explained below.

The $\mathscr{L}$ denotes the loss function, $y_{true}^{(i)}$ is the true label, $y_{pred}^{(i)}$ is the model predicted label, ϵ is a small value to avoid division by 0, which we keep at 1. N is the total number of pixels in an image i.e., 512*512, α is the weighing term used in Tversky loss, and γ is a modulating factor used in focal Tversky loss.

a. Dice loss: The Dice loss measures the overlap between the predicted segmentation mask and the ground truth by computing the harmonic mean of precision and recall. The Dice loss helps reduce class imbalance by penalizing false positives and false negatives.

$$\mathscr{L}\_(\text{Dice})\left(y_{true}^{(i)}, y_{pred}^{(i)}, \epsilon\right) = 1 - \frac{2\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \epsilon}{\sum_{i=1}^{N}\left(y_{pred}^{(i)}\right)^2 + \sum_{i=1}^{N}\left(y_{true}^{(i)}\right)^2 + \epsilon}$$

b. Jaccard loss: The Jaccard index, also known as Intersection Over Union (IOU), measures the intersection of the predicted flood mask to the ground truth mask. Since the objective of the segmentation models is to minimize the loss, Jaccard loss, formulated as 1-Jaccard index, is used. Jaccard loss is one of the most used loss functions for image segmentation tasks since it rewards the model for increasing the intersection between the predicted mask and output mask.

$$\mathscr{L}\_(\text{Jaccard})\left(y_{true}^{(i)}, y_{pred}^{(i)}, \epsilon, N\right) = 1 - \frac{\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \epsilon}{\sum_{i=1}^{N}\left(y_{pred}^{(i)}\right)^2 + \sum_{i=1}^{N}\left(y_{true}^{(i)}\right)^2 - \sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \epsilon}$$

c. Tversky loss: The Tversky loss is a generalization of the Dice and Jaccard losses. Tversky loss incorporates a weighting term allowing for uneven penalties for false positives and false negatives using the weighing term α. The Tversky loss performs well for imbalanced data and is more robust to noise and outliers. Although we aim to identify floods, we don't want to overestimate flood values, hence we set alpha at 0.7.

$$\mathscr{L}\_(\text{Tversky})\left(y_{true}^{(i)}, y_{pred}^{(i)}, \alpha, \epsilon, N\right) = 1 - \frac{\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \epsilon}{\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \alpha\sum_{i=1}^{N} y_{true}^{(i)}\left(1 - y_{pred}^{(i)}\right) + (1-\alpha)\sum_{i=1}^{N}\left(1 - y_{true}^{(i)}\right) y_{pred}^{(i)} + \epsilon}$$

d. Focal Tversky loss: Focal loss was proposed as a modulating factor to the general loss function to reduce the contribution of easy examples to the loss. The modulating factor is a function of the predicted probability of the correct class, which is high for well-classified examples and low for misclassified examples, thereby allowing the model to focus on hard examples. The Focal Tversky loss (Abraham and Khan, 2018) adds a focal parameter to the Tversky loss to penalize the model for incorrect predictions on the minority class, thereby developing a more accurate model for imbalanced data like floods. As for the Tversky loss the alpha is set at 0.7 γ which is the modulating factor for focal loss and is kept at 0.75.

$$\mathscr{L}\_(\text{FocalTversky})\left(y_{true}^{(i)}, y_{pred}^{(i)}, \alpha, \epsilon, N, \gamma\right) = \left(1 - \frac{\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \epsilon}{\sum_{i=1}^{N} y_{true}^{(i)} y_{pred}^{(i)} + \alpha\sum_{i=1}^{N} y_{true}^{(i)}\left(1 - y_{pred}^{(i)}\right) + (1-\alpha)\sum_{i=1}^{N}\left(1 - y_{true}^{(i)}\right) y_{pred}^{(i)} + \epsilon}\right)^{\gamma}$$

3.5. Model training and optimization

Our baseline model uses S1 SAR and auxiliary data as input, and outputs two logit bands which are converted into flood probability using SoftMax function. The models are trained for a maximum of 400 epochs using RMSprop optimizer with an initial learning rate of $1e^{-3}$. We monitor the F1 value on validation. The learning rate is reduced by a factor of 2, if the F1 score does not improve for 8 epochs. In case there is no improvement for 20 epochs, or the learning rate decreases to $1e^{-7}$ and the model stops training. We use minibatch gradient descent where the batch size varies from 12 to 24 based on model size (due to GPU size restriction large models have small batch sizes). To further improve our models, we use three other optimizations which are explained below.

3.5.1. Data augmentation

Data augmentation techniques are used to synthetically generate new data from the existing training data. It provides diversity in the training data, thereby increasing the model's ability to generalize and perform well on new, unseen data (Waldner et al., 2019). All the data augmentations used in this study are implemented using Albumentations library (Buslaev et al., 2020). In this study, we apply various image augmentation to the training data such as cropping, horizontal flip, vertical flip, and random rotation. The data augmentations are incorporated directly into the data generator with the probability of each augmentation set being between 0.1 and 0.3. This ensures that non-augmented data remains a predominant part of the training set, maintaining the integrity of the original dataset while still providing sufficient variation.

3.5.2. Multi-task learning setup

Multi-task learning (MTL) involves simultaneously performing multiple complementary tasks using shared representations, which allows models to leverage knowledge across different but related domains. In our case, we formulate the model to jointly predict the Modified Normalized Difference Water Index (MNDWI) alongside flood inundation mapping. The rationale behind using this formulation is that our labels are based on a well-defined threshold on the MNDWI. However, some studies have shown that the MNDWI threshold is not absolute, and MNDWI values between 0.1 and 0.3 can also indicate the presence of water (Du et al., 2016; Wen et al., 2021). Therefore, our MTL setup requires the model to predict both MNDWI values and binary labels. Given that MNDWI values indicating water presence can vary, the MNDWI prediction is treated as a regression problem with a threshold between 0 and 0.5. The regression task is optimized using a mean absolute error loss function. Our approach employs hard parameter sharing where all the parameters except the final layers are shared. To balance performance degradation and facilitate learning complementary information, we utilize a weighted loss function, where the weight for the segmentation loss is set at 0.9, while the weight for the scaled mean absolute error loss is set at 0.1. This configuration aims to optimize performance across both tasks effectively.

3.5.3. Model soups

Model soups average the weights of multiple trained models with same architecture to improve performance (Wortsman et al., 2022). The idea behind this averaging is similar to ensembles where the soup can

generalize better than single models. However, unlike ensembles there is no extra inference cost since the model architecture remains same. Model soups have recently achieved state of the art performance in classification tasks (Wortsman et al., 2022). Since we had already trained multiple models with same architecture but for different loss functions, we were able to develop model soups without additional training and no cost during inference. We use a greedy soup approach where all models are ranked by their score in decreasing order. The model with the best overall score serves as the base model (Wortsman et al., 2022). Subsequent models are added to this soup sequentially; if their inclusion improves accuracy on the validation set, their weights are incorporated into the model. This process continues until all models have been evaluated.

3.6. Deep ensembles

Model soups are great for optimizing along a single loss landscape; however, it is possible that the model reaches one of the many local minima. Consequently, a single deep learning model might excel at predicting certain subsets of data while underperforming on others. To address this issue, we propose developing deep learning models with diverse architectures and combining their predictions. While ensemble deep learning methods can increase inference costs, they also reduce overall variance and enhance prediction accuracy. In this study, we create a heterogeneous and diverse collection of deep learning models, and the ensembles serve the purpose of not only making predictions more robust, but also quantify uncertainty using deep ensembles.

A key step in ensemble development is member selection. In this study, we propose a novel gain algorithm for optimal deep learning model selection for ensemble generation from multiple deep learning models, such that an increase in model diversity is achieved without compromising the overall accuracy. The idea of the gain algorithm is to determine a set of models that can capture the overall distribution of training data effectively. This algorithm not only helps select optimal and diverse models but also identifies the number of models for ensemble generation. A detailed explanation of gain algorithm and its implementation are discussed below. We also provide a python implementation of gain algorithm.

a. Getting max accuracy per image and the corresponding model (Line 1–11): We first develop a database using training data which contains the accuracy metric (IOU) for every image for all the diverse trained models. Then we identify the highest value of accuracy metric for every image and label it as IOU_{max} and the corresponding model best model as IOU_{model}.
b. Weak model development (Line 12–16): We subset all the images with IOU_{max} < accuracy threshold (τ). These are called weak images since no model can perform well on them. We train a separate deep learning model called weak model using weak images. Since weak model has less and complex training data, we use multiple image augmentations on the 'weak images' to create a separate training database (D_S^{weak}).
c. Updating max accuracy and max accuracy model (Line 17–22): Since the weak model might produce better segmentation especially on weak images, we update the IOU_{max} and IOU_{model} with weak model data.
d. Removal of true weak images (Line 23–31): Even after updating the database with weak model, there are images on which none of the models perform adequately. These images are assumed to be noisy and their inclusion in final gain calculation might lead to incorrect ensemble model selection. Therefore, all images with $IOU_{max} < (\tau)$ are removed permanently from the database.
e. Base model selection and encoding (Line 32–33): The model which has the highest accuracy for the maximum number of images is selected as the base model ($Model_{base}$) and the corresponding base model accuracy for every image is (IOU_{base}). We also develop the one hot encoding of the (IOU_{model}) as (IOU_{OHE}).
f. Residue and gain calculation (Line34:49): We define residue as IOU_{max} -IOU_{base} and calculate the residue for every image. The residue aims to quantify the improvement over base model is available for that image. The residue is multiplied with IOU_{OHE} and we calculate the sum of all IOU_{OHE}. This is called gain list and shows the information every model adds over the base model. The max value of gain list gives the maximum information gain, and the argmax of the gain column gives the model which adds the most information over the base model, the selected model is the first ensemble member. This process is repeated for all iterations with the new definition of residue being IOU_{max}-max(IOU_{base}, ….,IOU_j) where j is the iteration number.
g. Information change calculation and ensemble model selection (Line49:54): We calculate the gain percentage from the gain list. If there isn't a substantial information gain by incorporating the new model, the algorithm terminates, and we discard all the other models. The minimum information gain for a model to become and ensemble member is defined by the user.

After model selection, ensemble results are computed using the soft voting technique which uses the mean of probabilities of individual ensemble members. Soft voting ensembles are used because they are fast and do not require the models to be homogeneous, therefore they have been effective in a wide range of applications, including image recognition, natural language processing, and time series forecasting (Salur and Aydın, 2022; Tasci et al., 2021).

3.7. Uncertainty quantification

Quantification of uncertainty in model predictions helps increase confidence in its predictions and avoid poor decision making. Although deep learning models have high accuracy, they often struggle with overconfident predictions, making them poor at quantifying uncertainty. Traditionally Bayesian Deep Neural Networks have been used to predict uncertainty by learning a distribution over model weights, however, they are computationally expensive and training them requires significant modifications to the architecture. In this study we use deep ensembles to quantify data uncertainty following the approach proposed by (Lakshminarayanan et al., 2017). Deep ensembles are based on the theory that given a finite training data, numerous model configurations can fit the observations with comparable accuracy and capture a range of viable solutions. We formulate the uncertainty based on the probability estimates of individual models. Assuming there are n ensemble members and P_i is the per pixel probability for the ith model, the uncertainty δ associated with every pixel is defined as

$$\delta = \left(\frac{2}{n}\sum_{i=1}^{n} P_i\right) \text{ if } \sum_{i=1}^{n}\frac{P_i}{n} < 0.5\ (\text{label} = 0) \tag{3}$$

$$\delta = \left(2 - \frac{2}{n}\sum_{i=1}^{n} P_i\right) \text{ if } \sum_{i=1}^{n}\frac{P_i}{n} > 0.5\ (\text{label} = 1) \tag{4}$$

3.8. Evaluation metrics

For a binary segmentation problem based on the predicted label and true label, there are four possible outcomes.

a. The flood pixels correctly classified are True Positives (TP).
b. The non-flood pixels correctly classified are True Negatives (TN).
c. The flood pixels incorrectly classified are False Negatives (FN).
d. The non-flood pixels incorrectly classified are False Positives (FP).

The most used metric for model testing is accuracy, however accu-

racy can be misleading in the case of severely imbalanced data such as floods. Therefore, we use metrics which can effectively capture how well a model can map floods, such as F1, sensitivity, precision, and intersection over union (IOU).

a. Precision and Sensitivity: Precision measures the proportion of correctly identified flooded areas among all areas predicted as flooded by a model. It is quantified as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

A high precision indicates that the model has a lower number of false positives i.e pixels incorrectly mapped as floods. However, precision alone does not provide a complete picture of model accuracy. Sensitivity measures how well a model can identify true flooded pixels out of all the flooded pixels. Sensitivity is quantified as:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

A model with high precision but low sensitivity might underestimate floods, while one with high sensitivity but low precision might overestimate them, therefore balancing precision and sensitivity is crucial for developing deep learning models for flood detection.

b. F1 Score: The F1 score computes the harmonic mean of precision and sensitivity, providing a balanced assessment of the model's performance. The F1 score is calculated as:

$$\text{F1 score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

It effectively captures the model's ability to detect floods while accounting for both false positives and false negatives. This makes the F1 score particularly valuable in class imbalance scenarios like flood mapping.

c. IOU Score: The Intersection over Union (IOU) score quantifies the overlap between predicted flood regions and their corresponding ground truth. It is calculated as:

$$\text{IOU score} = \frac{TP}{TP + FP + FN}$$

IOU like F1 score is particularly valuable for evaluating flood segmentation models because it is less sensitive to class imbalance. This metric has been widely used in various imbalanced quantifications for natural hazards, including floods and landslides. Due to its benefits IOU score has been used in multiple imbalanced natural hazards mapping problems like floods and landslides (Sharma et al., 2024; Sharma and Saharia, 2024; Yu et al., 2023).

By using Precision, Sensitivity, F1 score, and IOU score in combination we can ensures that models are not only accurate but also reliable for operational flood monitoring applications.

3.9. Open-source tool

We provide an open-source, fully automated tool, DeepSARFlood developed using python 3.9 and Pytorch which demonstrates the implementation of our proposed framework. DeepSARFlood requires only two user inputs: the date of occurrence of flood, and the coordinates of the area of interest. The overall process of the module is represented in Fig. 1 section b. The detailed explanation of its working is explained below.

a) Data Fetching: We use Google Earth Engine (GEE) to fetch input datasets. The S1 SAR data from GEE is provided as preprocessed tiles in decibels (dB) with a 10 m spatial resolution, including thermal noise removal, radiometric calibration, and terrain correction using a DEM (Gorelick et al., 2017). DeepSARFlood is fully parallelized using joblib (https://joblib.readthedocs.io/en/latest/) for data fetching and data extraction.

b) Preprocessing: After the data fetching the data, each band is normalized in the values between 0 and 1 like the training process. The bands are then merged to create a stack.

c) Patch development: The input dataset is of arbitrary size however our model is trained on patches of size 512*512. We use a sliding window technique where a window of size 512*512 is moved over the input data patch with a stride of 400. A stride lower than window size makes sure that pixels on the edges of window which cannot see neighbor data are accounted for in next windows.

d) Model Run: After the development of patches, DeepSARFlood instantiates the model and loads the number of patches equivalent to the batch size into the GPU to generate flood probabilities. Due to overlapping patches, edge pixels possess multiple probability values. To maintain uniformity, we sum all probability values and keep a count of the number of times each probability is updated. The final probability map is generated by dividing the cumulative probability by the count. DeepSARFlood computes the probability for all ensemble members and subsequently, it develops the flood inundation layer where the ensemble probability exceeds 0.5. It further constructs the uncertainty layer using equations described in Section 3.7. For a deeper understanding of the working of our models, users can also use a probability setup, which provides individual model probabilities along with ensemble outputs.

e) Post Processing: Before converting the final predictions to flood maps, all areas with a slope greater than 5% are converted to non-flooded, since high slope pixels drain water to pixels with lower elevations (Tazmul Islam and Meng, 2022). The final dataset is geotagged and provided as a GeoTiff file for easy visualization in any GIS software.

f) Visualization: The aim of DeepSARFlood is to visualize flooding. To separate permanent water and flooded areas DeepSARFlood uses JRC global surface water layer. The area which has a seasonality greater than 5 (implying the area has water for more than 5 months) is considered permanent water and excess water is considered flood.

DeepSARFlood encapsulates the algorithms developed as a part of this study to provide a user-friendly interface for flood inundation. The DeepSARFlood is modular, open source, and flexible, and contains multiple functions for visualization of outputs as well as benchmarking with S2 data.

4. Experimentation and results

4.1. Training details

All experiments were carried out on a workstation with an intel i9 14900 KF processor and NVIDIA RTX 4090 with 24 GB VRAM and 128 GB RAM. Our initial experiments consisted of developing CNN based encoders. We performed an exhaustive search using various CNN based encoders, including ResNet50, ResNet18, ResNeXt50, EfficientNetB0, EfficientNetB3, EfficientNetB7, and InceptionV3. Additionally, we tested multiple decoders for the CNNs, namely DeepLabV3+, U-Net, and U-Net++. However, U-Net++ significantly outperforms the other decoders, leading us to discontinue the use of other decoders in subsequent experiments.

We then explored ViT-based and hybrid encoders, such as ViT, Swinv2, MaxViT, and CoAtNet, which significantly outperform the CNN-based encoders. We only use the small and base versions of ViTs and don't-test the large ViT encoders due to limited data availability, computational requirements and good score provided by small and base encoders. Although larger models might offer better accuracy, they require a much larger dataset to be trained well. Also, DeepSARFlood is meant to be an operational flood mapping tool with ensemble

Table 1
Results on Sen1Floods11 test data with various optimizations used in this study.

Model	IOU
Base (U-Net Swin-V2)	0.682
Base + Augmentations	0.679
Base + MTL	0.712
Base + MTL + model soups	0.722

Table 2
Comparison of DeepSARFlood with similar SAR-based flood mapping studies.

Model	mIOU	Reference	Developer	Training Dataset
U-Net + Swin-V2 (MTL + model soups)	0.722	Proposed	IIT Delhi	Sen1Floods11 + custom weak labels
DeepLabV3+ inception 65 (cross model distillation)	0.727	Garg et al. (2023)	Google	Sen1Floods11 + Floods208
U-Net + EfficientNetB0	0.556	Wieland et al. (2024)	German Aerospace Center (DLR)	S1S2-Water
BASNet	0.5407	Bai et al. (2021)	Renmin University	Sen1Floods11

Table 3
Ensemble and individual model performance on Sen1Floods11 test set.

Model	IOU	F1	Precision	Recall
U-Net Swin-V2-base	0.7226	0.7891	0.7934	0.8669
U-Net++ ResNet50	0.6940	0.7682	0.7896	0.8346
U-Net MaxViT-base	0.6910	0.7572	0.7591	0.8590
Ensemble	0.7153	0.7816	0.7930	0.8541

uncertainties and using large ViT models would increase the runtime limiting its usage.

4.2. Model optimization

As explained in Section 3 for a single model optimization, we use data augmentation, MTL and model soups to improve our models. Although all these optimizations are used for every model the statistics in Table 1 are provided for the best performing model (Swin-v2 encoder with U-Net decoder) on Sen1Floods11 test dataset.

Data augmentation has a negative impact on validation loss as compared to baseline, hence they were dropped in further experiments. The MTL has a significant impact improving the base mIOU of model from 0.68 to 0.712 on the test data. Adding the model soups further increased the mIOU from 0.712 to 0.722. Recent studies have reported various accuracy levels on the Sen1Floods11 test dataset, utilizing both classical thresholding techniques and CNN-based methods. Table 2 provides an overview of how our study compares with similar methods. Notably our model outperforms most other methods, and the accuracy is comparable to that achieved using cross model distillation approach for flood mapping (Garg et al., 2023). Although the approach of multi-task learning presented in this study and cross model distillation are different, the core idea remains similar. Both studies demonstrate that while developing flood inundation models using S1 data with weak labels from S2, the model can benefit from additional information transfer during training, thereby resulting in performance improvements (see Table 3).

4.3. Ensembles

The base model is Swin-v2 encoder with U-Net decoder and the ensemble members are MaxViT encoder with U-Net decoder and Resnet50 encoder with U-Net++ decoder. Although all models have similar performance on validation set, Swin-V2 has a much better performance in the test set. The ensemble has a lower but equivalent performance on the test set. In practical applications (see section 5), the ensemble model outperforms all the other models. Moreover, ensembles also help us develop real-time uncertainty estimates.

Table 4
Deep learning model runtime for DeepSARFlood.

Area size	Runtime (seconds)
0.1^o x 0.1^o	1.2
0.5^o x 0.5^o	12.1
1^o x 1^o	40.8
2^o x 2^o	158.3

Table 5
Comparison of model performances for Pakistan floods 2022 using Sentinel 2 labels.

Encoder	IOU	F1	Precision	Recall
U-Net Swin-V2-base	0.7305	0.8442	0.7818	0.9175
U-Net++ ResNet50	0.7086	0.8294	0.8012	0.8598
U-Net MaxViT-base	0. 7229	0.8392	0.7666	0.9270
Ensemble	0.7410	0.8513	0.7917	0.9205

4.4. Runtime

Another important factor in the real-time deployment of deep learning models is runtime. Table 4 provides a benchmark runtime excluding the data fetch time (all the runtime experiments are carried out using the workstation specifications mentioned earlier and include model loading time). Although the data fetching is automatic in the provided tool, the data fetching time depends on the internet speed and varies, hence that time is excluded.

The model runs comparatively faster and can provide output for 1^o x 1^o (12100 km^2) area in approximately 40 s. This rapid processing capability, combined with the model's high accuracy and robust uncertainty estimation, underscores its suitability for real-time deployment.

5. Case study

We present two case studies to assess how DeepSARFlood performs in real world, and the results are compared with the flood maps developed using S2 imagery.

5.1. Pakistan floods (2022)

The devastating Pakistan floods occurred from July–September 2022, were caused by excessive rainfall and resulting in damages estimated at 30 billion USD (Hong et al., 2023). Since the combination of concurrent cloud-free S2 and S1 SAR images is rare, we could only get a small cloud-free area. To develop SAR based flood inundation maps, we set the date to August 06, 2022 and generate a flood layer from S2 data. Table 5 shows statistics of individual and ensemble models considering S2 data as true label (see Fig. 3).

As shown in Table 5, U-Net Swin-v2 achieved the highest overall IOU and F1 score. However, the models identified by the gain algorithm have complementary strengths. U-Net++ ResNet50 exhibited higher precision and lower recall, indicating it is better at identifying non-flood areas but less effective at detecting flooded areas. Conversely, MaxViT showed lower precision and higher recall, meaning it is better at predicting flooded areas. This complementary behavior demonstrates the diversity of the ensembles identified by gain algorithm, which

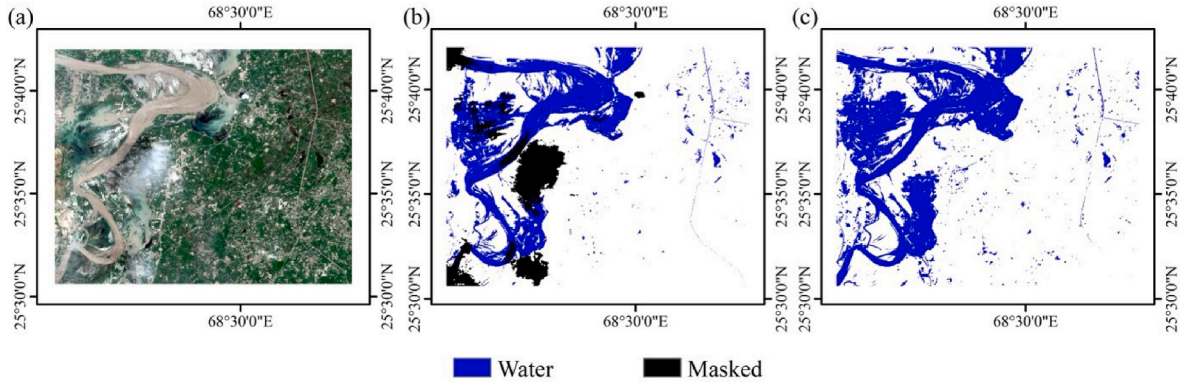

Fig. 4. Comparison of flood layer generated using Sentinel 2 data and DeepSARFlood for 2022 Pakistan floods. a) RGB visualization of Sentinel 2 b) Flood layer developed using Sentinel 2 data. c) DeepSARFlood flood layer.

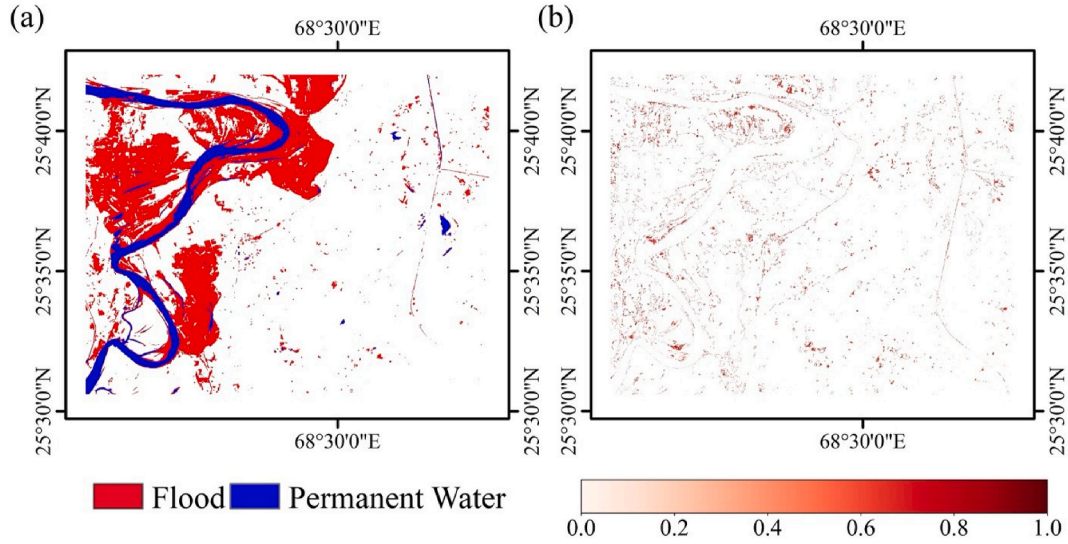

Fig. 5. Flood and uncertainty layers for 2022 Pakistan floods. a) Flood layer developed using DeepSARFlood b) Uncertainty layer of DeepSARFlood.

collectively outperform individual models with an IOU of 0.7410 and an F1 score of 0.8513. The ensemble model also has a high recall of 0.92 in comparison to a precision of 0.79 showing that DeepSARFlood has relatively large false positives than false negatives. This is significant since during mapping natural hazards such as floods an underestimation of the impact is more detrimental than overestimation. Fig. 4 shows flood maps developed using DeepSARFlood and S2 data. Although both show similar areas as flooded as is evident from statistics, a significant portion of S2 is masked since it is covered with clouds.

Fig. 5 ashow the permanent water in blue and flooded area in red. We use the JRC permanent water to delineate permanent water layers and the rest of inundated water is considered as floods. Fig. 5 b shows the uncertainty layer. We can see that most models have similar flooded locations, however there is large uncertainty along the edges (see Fig. 6).

5.2. Assam (India) floods (2022)

The Brahmaputra River is one of the most important riven basins in India known for its annual recurring floods particularly in the Indian state of Assam (Singh et al., 2024). The primary cause of the 2022 floods, which began in May and lasted until October, was heavy monsoon rainfall. We aim to maximize the non-cloudy area of the S2 imagery develop the flood extent, therefore we set the area of interest between 93.00° and 94.00° longitude and 26.60°–26.90° latitude and the date to August 10, 2022. Table 6 shows the statistics considering S2 flood map as true labels.

The results of the ensemble model are consistent with those observed in the first case study. The individual models retain their strength with U-Net++ ResNet50 having a better overall precision and U-Net MaxViT-base having a better recall. However, contrary to the findings from the test data and the Pakistan floods case study, the MaxViT encoder with a U-Net decoder demonstrates superior overall performance compared to the base model (Swin-V2 encoder with a U-Net decoder). These findings further highlight that employing a diverse ensemble of models is more effective in reducing overall uncertainty, indicating that a single model may not be suitable for all use cases.

6. Discussion

Mapping floods using Synthetic Aperture Radar (SAR) data presents unique challenges, primarily due to the difficulty in developing flood labels directly from SAR imagery. Historically, flood labels have been generated by using concurrent SAR and optical imagery, with manual labeling of optical images serving as flood labels for SAR data. However, this approach is time-consuming and costly(Bonafilia et al., 2020). This study uses recent advancements in cloud computing, artificial intelligence, and cloud removal techniques for a more efficient framework to develop high-quality weak labels and train optimized deep learning models for flood mapping. The key improvements over historical works are discussed below.

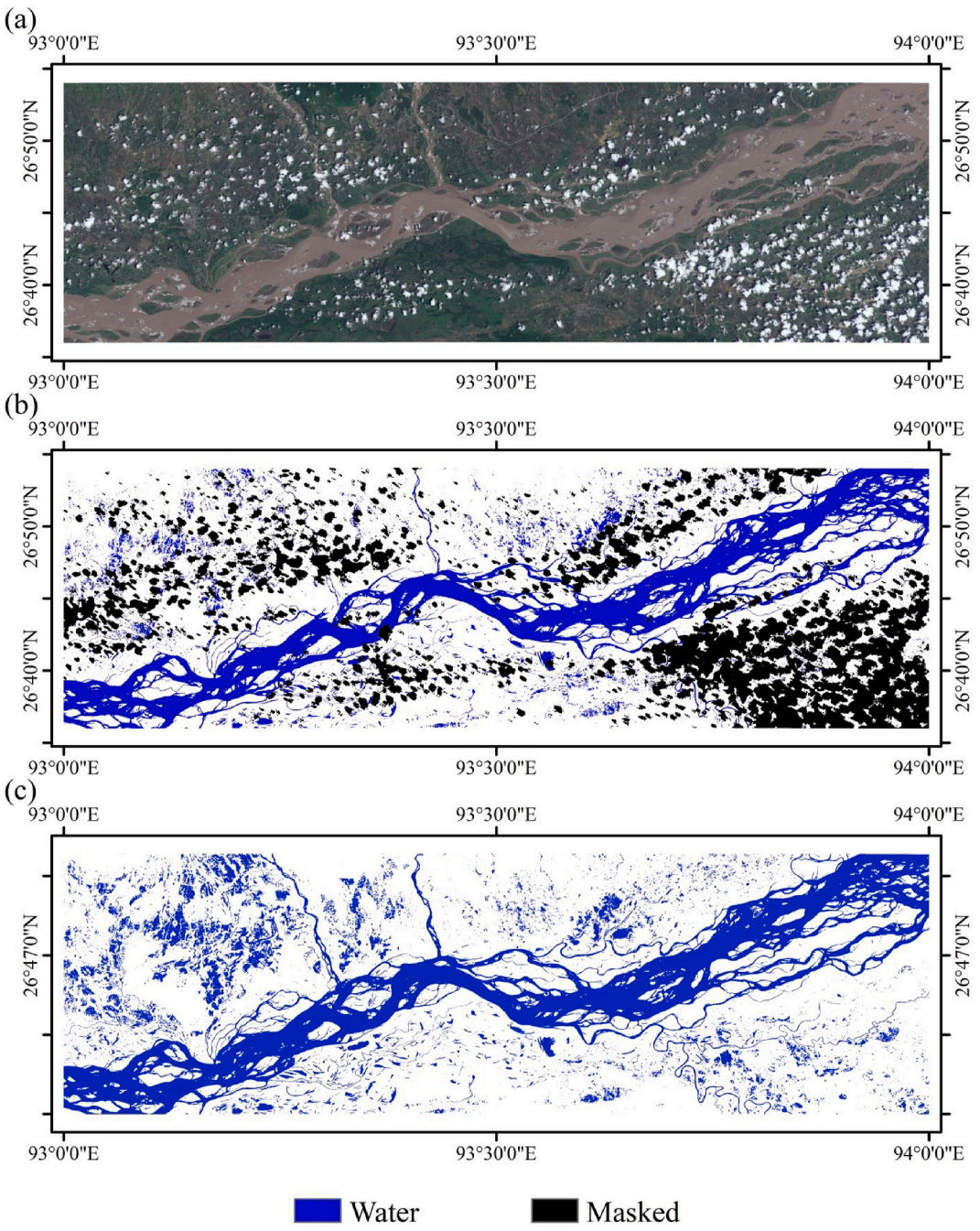

Fig. 6. Comparison of DeepSARFlood with Sentinel 2 data. a) RGB visualization of Sentinel 2 image b) Flood map using S2 c) Flood mapped using DeepSARFlood.

a. Automated Label Generation: We introduce an improved weak-labelling approach that leverages concurrent Sentinel-1 and Sentinel-2 imagery to develop flood labels. This method significantly reduces the need for expensive human annotation while enhancing label quality. By increasing the availability of reliable training data, the framework contributes to the development of more robust and generalizable models as shown in the results.
b. Improved Accuracy and Robustness: We use an ensemble of Vision Transformer (ViT) and Convolutional Neural Network (CNN) based models to learn complex patterns related to floods in SAR data, resulting in substantial accuracy improvements. We optimize individual deep learning models using multi-task learning and model soups and improve diversity using novel ensemble based techniques. This approach creates a balance of high-precision and high-recall models, leading to improved overall accuracy. The models have demonstrated impressive performance, achieving an Intersection over Union (IoU) score of 0.72 on benchmark datasets like Sen1-Floods11. By combining outputs from multiple deep learning architectures, the framework enhances both accuracy and robustness in predictions, reducing variance and improving performance.
c. Efficient Use of Computational Resources: The deployment method uses a sliding window technique which splits the images into patches and uses a stride lower than window size ensuring that edge pixels are accounted for in subsequent windows. The patches are loaded into GPU with optimized batch sizes based on model size (large models of ensemble have low batch size). To maintain optimal GPU performance for each model, cache clearing and garbage collection are employed after every model run. This formulation ensures that computational resources are used efficiently across GPUs.

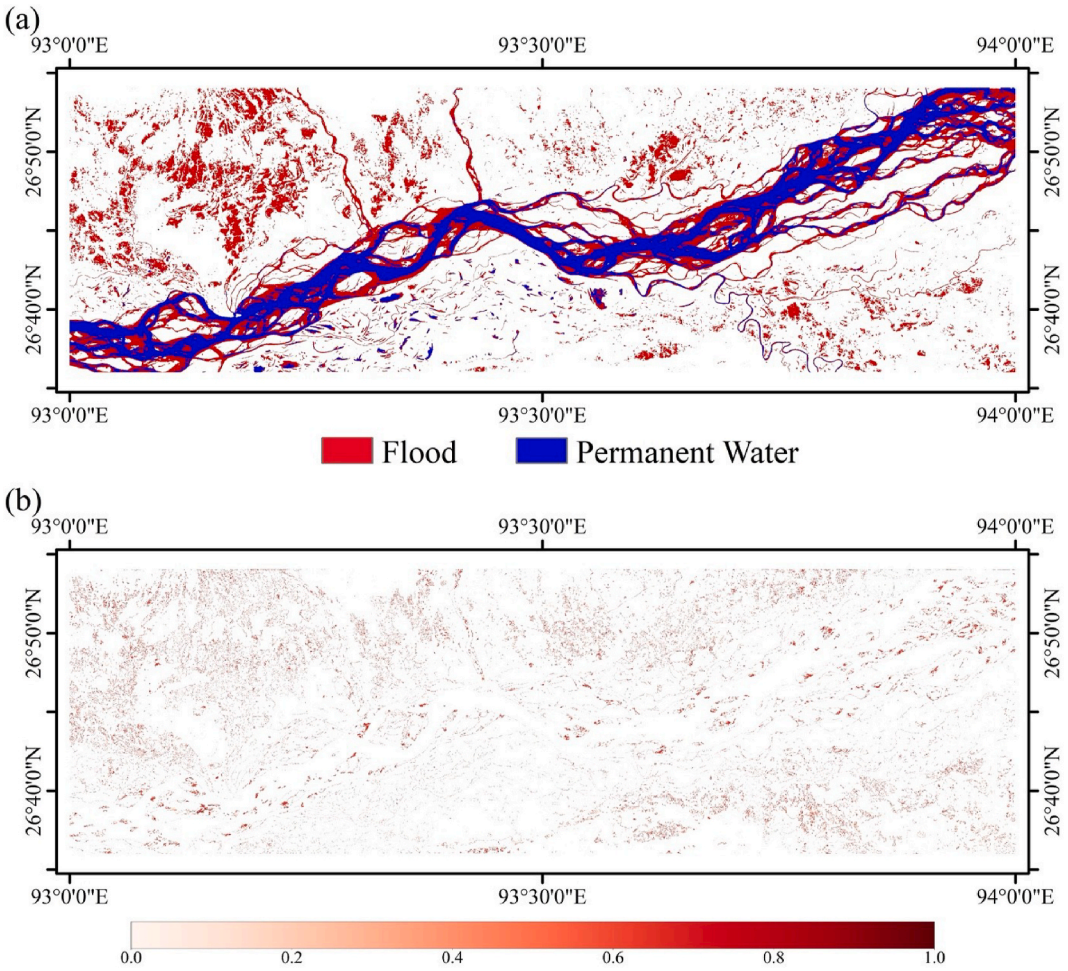

Fig. 7. Flood and uncertainty layers for 2022 Brahmaputra floods dated August 10, 2022. a) Flood layer developed using DeepSARFlood) b) Uncertainty layer of DeepSARFlood.

Table 6
Comparison of model performances for Brahmaputra floods 2022 using Sentinel 2 labels.

Models	IOU	F1	Precision	Recall
U-Net Swin-V2-base	0.7364	0.8482	0.7688	0.9459
U-Net++ ResNet50	0.7356	0.8476	0.7755	0.9346
U-Net MaxViT-base	0.74338	0.8528	0.7743	0.9489
Ensemble	0.7478	0.8557	0.7795	0.9482

d. Operational tool for real time disaster management: We have developed an operational tool with a user-friendly GUI interface, requiring only the area of interest and flood date as inputs. The ease of use makes it accessible to end-users without the need for backend knowledge. DeepSARFlood also features multiple visualization modules allowing novice users to interpret the results effectively. The DeepSARFlood tool aims to democratize flood mapping by offering ease of use, speed, and accuracy, making it suitable for real-time scenarios.

However, this study suffers from a few challenges which will form the basis for improvements.

a. Fixed permanent water layer: For ease of use we deploy a fixed permanent water layer based on the JRC data which may not accurately capture the evolving courses of rivers. To address this, a change module has been introduced, enabling users to specify a date outside the flood period whose inundation extent serves as the baseline for permanent water. The presence of additional presence of water beyond that baseline is classified as flood inundation.
b. Usability in urban areas: In urban settings, the presence of tall buildings often causes double bounce scattering, making it difficult to identify flooding with only GRD data. The current models are trained and validated on riverine floods. Even the test data developed from Sen1Floods11 contains only riverine data. Hence this model cannot work for mapping floods in urban areas. In future to improve the DeepSARFlood algorithm we aim to use InSAR coherence data to improve flood mapping in urban regions.
c. Open-Source data: The current study aims to provide an open-source framework for broader accessibility. With the system failure of S1B (since December 23, 2021), only S1A provides open-source real-time SAR data. Upcoming missions such as NISAR and Sentinel 1-C will provide valuable new datasets. Despite SAR datasets having the same modality the deep learning models trained on S1 data cannot be directly applied to other band SAR data and will require domain adaptation to make them useful.

7. Conclusion

Mapping floods from SAR data is a complex task. The primary issues include the complexity of interpreting SAR imagery, limited availability of high-quality labeled data for training deep learning models. In order to overcome these problems, we develop an end-to-end framework for generating SAR-based flood inundation maps using weak labels derived

from concurrent optical imagery. By leveraging multi-task learning and model soups, we are able to develop optimized models that effectively utilize weakly labeled data. Additionally, we introduce novel gain algorithm to create diverse deep ensembles, enhancing the accuracy and developing uncertainty quantification. This approach addresses the challenges posed by the limited availability of hand-annotated flood data, resulting in state-of-the-art flood inundation mapping capabilities. Our framework not only achieves an IoU of 0.7226 on Sen1Floods11 benchmark datasets but also demonstrates usage in real-time flood inundation mapping with an IoU of 0.7478 for Assam floods and 0.7410 for Pakistan floods. The speed of our model allows for deployment in real-time applications, making it a valuable tool for emergency response. However, there are a few limitations which will form the basis of future development. Firstly, this model is trained in riverine floods and not suited for urban floods. Secondly, we rely on open source S-1 data which has a low temporal resolution, hindering its usage in real-time. The outcomes of this study hold significant potential for improving flood risk assessment, optimizing resource allocation during flood events, supporting insurance claims processing, facilitating the development of historical flood inundation archives, and assimilating of satellite-based flood outputs in numerical models.

CRediT authorship contribution statement

Nirdesh Kumar Sharma: Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Manabendra Saharia:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Compliance with Ethical standards

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

- Training dataset (Sen1Floods11): https://github.com/cloudtostreet/Sen1Floods11
- DeepSARFlood Tool: https://github.com/hydrosenselab/DeepSARFlood

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.

Acknowledgements

This research was conducted in the HydroSense lab of IIT Delhi and the authors acknowledge the IIT Delhi High Performance Computing facility for providing computational and storage resources. Dr. Manabendra Saharia gratefully acknowledges financial support for this work through grants from Ministry of Earth Sciences/IITM Pune Monsoon Mission III (RP04574); Ministry of Earth Sciences (RP04741); and DST IC-IMPACTS (RP04558).

Data availability

Data is open

References

Abraham, N., Khan, N.M., 2018. A Novel Focal Tversky Loss Function with Improved Attention U-Net for Lesion Segmentation.

AghaKouchak, A., Chiang, F., Huning, L.S., Love, C.A., Mallakpour, I., Mazdiyasni, O., Moftakhari, H., Papalexiou, S.M., Ragno, E., Sadegh, M., 2020. Climate Extremes and Compound hazards in a Warming world. Annu. Rev. Earth Planet Sci. 48, 519–548. https://doi.org/10.1146/annurev-earth-071719-055228.

Amitrano, D., Di Martino, G., Iodice, A., Riccio, D., Ruello, G., 2018. Unsupervised rapid flood mapping using Sentinel-1 GRD SAR images. IEEE Trans. Geosci. Remote Sens. 56, 3290–3299. https://doi.org/10.1109/TGRS.2018.2797536.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39, 2481–2495. https://doi.org/10.1109/TPAMI.2016.2644615.

Bai, Y., Wu, W., Yang, Z., Yu, J., Zhao, B., Liu, X., Yang, H., Mas, E., Koshimura, S., 2021. Enhancement of detecting permanent water and temporary water in flood disasters by Fusing Sentinel-1 and Sentinel-2 imagery using deep learning algorithms: Demonstration of Sen1Floods11 benchmark datasets. Remote Sens 13, 2220. https://doi.org/10.3390/rs13112220.

Benoudjit, A., Guida, R., 2019. A novel fully automated mapping of the flood extent on SAR images using a supervised classifier. Remote Sens 11, 779. https://doi.org/10.3390/rs11070779.

Bereczky, M., Wieland, M., Krullikowski, C., Martinis, S., Plank, S., 2022. Sentinel-1-Based water and flood mapping: benchmarking convolutional neural networks against an operational Rule-based processing chain. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 15, 2023–2036. https://doi.org/10.1109/JSTARS.2022.3152127.

Bischke, B., Helber, P., Borth, D., Dengel, A., 2018. Segmentation of imbalanced Classes in satellite imagery using adaptive uncertainty weighted class loss. In: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. Presented at the IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, pp. 6191–6194. https://doi.org/10.1109/IGARSS.2018.8517836.

Bonafilia, D., Tellman, B., Anderson, T., Issenberg, E., 2020. Sen1Floods11: a georeferenced dataset to train and test deep learning flood algorithms for Sentinel-1. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Seattle, WA, USA, pp. 835–845. https://doi.org/10.1109/CVPRW50498.2020.00113.

Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A., 2020. Albumentations: fast and flexible image augmentations. Information 11, 125. https://doi.org/10.3390/info11020125.

Chen, H., Qi, Z., Shi, Z., 2022. Remote sensing image change detection with transformers. IEEE Trans. Geosci. Remote Sens. 60, 1–14. https://doi.org/10.1109/TGRS.2021.3095166.

Clement, M.a., Kilsby, C.g., Moore, P., 2018. Multi-temporal synthetic aperture radar flood mapping using change detection. J. Flood Risk Manag. 11, 152–168. https://doi.org/10.1111/jfr3.12303.

Dechesne, C., Lassalle, P., Lefèvre, S., 2021. Bayesian deep learning with Monte Carlo dropout for Qualification of semantic segmentation. In: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. Presented at the 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, pp. 2536–2539. https://doi.org/10.1109/IGARSS47720.2021.9555043.

DeVries, B., Huang, C., Armston, J., Huang, W., Jones, J.W., Lang, M.W., 2020. Rapid and robust monitoring of flood events using Sentinel-1 and Landsat data on the Google Earth Engine. Remote Sens. Environ. 240, 111664. https://doi.org/10.1016/j.rse.2020.111664.

Donchyts, G., Winsemius, H., Schellekens, J., Erickson, T., Gao, Hongkai, Savenije, H., Giesen, N.V.D., 2016. Global 30m height above the nearest drainage. https://doi.org/10.13140/RG.2.1.3956.8880.

Du, Y., Zhang, Y., Ling, F., Wang, Q., Li, W., Li, X., 2016. Water bodies' mapping from Sentinel-2 imagery with modified normalized difference water index at 10-m spatial resolution produced by Sharpening the SWIR band. Remote Sens 8, 354. https://doi.org/10.3390/rs8040354.

Feng, Q., Gong, J., Liu, J., Li, Y., 2015. Flood mapping based on multiple Endmember spectral Mixture analysis and random forest Classifier—the case of Yuyao, China. Remote Sens 7, 12539–12562. https://doi.org/10.3390/rs70912539.

Fort, S., Hu, H., Lakshminarayanan, B., 2020. Deep ensembles: a loss landscape Perspective. https://doi.org/10.48550/arXiv.1912.02757.

Garg, S., Feinstein, B., Timnat, S., Batchu, V., Dror, G., Rosenthal, A.G., Gulshan, V., 2023. Cross-modal distillation for flood extent mapping. Environ. Data Sci. 2, e37. https://doi.org/10.1017/eds.2023.34.

Giustarini, L., Hostache, R., Kavetski, D., Chini, M., Corato, G., Schlaffer, S., Matgen, P., 2016. Probabilistic flood mapping using synthetic aperture radar data. IEEE Trans. Geosci. Remote Sens. 54, 6958–6969. https://doi.org/10.1109/TGRS.2016.2592951.

Gleason, C.J., Hamdan, A.N., 2017. Crossing the (watershed) divide: satellite data and the changing politics of international river basins. Geogr. J. 183, 2–15. https://doi.org/10.1111/geoj.12155.

Goffi, A., Stroppiana, D., Brivio, P.A., Bordogna, G., Boschetti, M., 2020. Towards an automated approach to map flooded areas from Sentinel-2 MSI data and soft integration of water spectral features. Int. J. Appl. Earth Obs. Geoinformation 84, 101951. https://doi.org/10.1016/j.jag.2019.101951.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R., 2017. Google earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sens. Environ. https://doi.org/10.1016/j.rse.2017.06.031.

Hamidi, E., Peter, B.G., Muñoz, D.F., Moftakhari, H., Moradkhani, H., 2023. Fast flood extent monitoring with SAR change detection using google earth Engine. IEEE Trans. Geosci. Remote Sens. 61, 1–19. https://doi.org/10.1109/TGRS.2023.3240097.

He, X., Zhou, Y., Zhao, J., Zhang, D., Yao, R., Xue, Y., 2022. Swin transformer Embedding UNet for remote sensing image semantic segmentation. IEEE Trans. Geosci. Remote Sens. 60, 1–15. https://doi.org/10.1109/TGRS.2022.3144165.

Hertel, V., Chow, C., Wani, O., Wieland, M., Martinis, S., 2023. Probabilistic SAR-based water segmentation with adapted Bayesian convolutional neural network. Remote Sens. Environ. 285, 113388. https://doi.org/10.1016/j.rse.2022.113388.
Hong, C.-C., Huang, A.-Y., Hsu, H.-H., Tseng, W.-L., Lu, M.-M., Chang, C.-C., 2023. Causes of 2022 Pakistan flooding and its linkage with China and Europe heatwaves. Npj Clim. Atmospheric Sci. 6, 1–10. https://doi.org/10.1038/s41612-023-00492-2.
Isikdogan, F., Bovik, A.C., Passalacqua, P., 2017. Surface water mapping by deep learning. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 10, 4909–4918. https://doi.org/10.1109/JSTARS.2017.2735443.
Jiang, X., Liang, S., He, X., Ziegler, A.D., Lin, P., Pan, M., Wang, D., Zou, J., Hao, D., Mao, G., Zeng, Y., Yin, J., Feng, L., Miao, C., Wood, E.F., Zeng, Z., 2021. Rapid and large-scale mapping of flood inundation via integrating spaceborne synthetic aperture radar imagery with unsupervised deep learning. ISPRS J. Photogramm. Remote Sens. 178, 36–50. https://doi.org/10.1016/j.isprsjprs.2021.05.019.
Klemas, V., 2015. Remote sensing of floods and flood-prone areas: an overview. J. Coast Res. 31, 1005–1013. https://doi.org/10.2112/JCOASTRES-D-14-00160.1.
Konapala, G., Kumar, S.V., Khalique Ahmad, S., 2021. Exploring Sentinel-1 and Sentinel-2 diversity for flood inundation mapping using deep learning. ISPRS J. Photogramm. Remote Sens. 180, 163–173. https://doi.org/10.1016/j.isprsjprs.2021.08.016.
Lakshminarayanan, B., Pritzel, A., Blundell, C., 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In: Advances in Neural Information Processing Systems. Curran Associates, Inc.
Li, S., Sun, D., Yu, Y., 2013. Automatic cloud-shadow removal from flood/standing water maps using MSG/SEVIRI imagery. Int. J. Remote Sens. 34, 5487–5502. https://doi.org/10.1080/01431161.2013.792969.
Li, Y., Martinis, S., Plank, S., Ludwig, R., 2018. An automatic change detection approach for rapid flood mapping in Sentinel-1 SAR data. Int. J. Appl. Earth Obs. Geoinformation 73, 123–135. https://doi.org/10.1016/j.jag.2018.05.023.
Li, Z., Demir, I., 2023. U-net-based semantic classification for flood extent extraction using SAR imagery and GEE platform: a case study for 2019 central US flooding. Sci. Total Environ. 869, 161757. https://doi.org/10.1016/j.scitotenv.2023.161757.
Manavalan, R., 2017. SAR image analysis techniques for flood area mapping - literature survey. Earth Sci. Inform. 10, 1–14. https://doi.org/10.1007/s12145-016-0274-2.
Manjusree, P., Prasanna Kumar, L., Bhatt, C.M., Rao, G.S., Bhanumurthy, V., 2012. Optimization of threshold ranges for rapid flood inundation mapping by evaluating backscatter profiles of high incidence angle SAR images. Int. J. Disaster Risk Sci. 3, 113–122. https://doi.org/10.1007/s13753-012-0011-5.
Muszynski, M., Hölzer, T., Weiss, J., Fraccaro, P., Zortea, M., Brunschwiler, T., 2022. Flood event detection from Sentinel 1 and Sentinel 2 data: does land Use Matter for performance of U-net based flood Segmenters?. In: 2022 IEEE International Conference on Big Data (Big Data). Presented at the 2022 IEEE International Conference on Big Data (Big Data), pp. 4860–4867. https://doi.org/10.1109/BigData55660.2022.10020911.
Nagel, G., Darby, S., Leyland, J., 2023. A global analysis of the timing of changes in water extents using Google Earth Engine and Landsat Time Series. (No. EGU23-4438). Presented at the EGU23, Copernicus Meetings. https://doi.org/10.5194/egusphere-egu23-4438.
Nobre, A.D., Cuartas, L.A., Hodnett, M., Rennó, C.D., Rodrigues, G., Silveira, A., Waterloo, M., Saleska, S., 2011. Height above the Nearest Drainage – a hydrologically relevant new terrain model. J. Hydrol. 404, 13–29. https://doi.org/10.1016/j.jhydrol.2011.03.051.
Pasquarella, V.J., Brown, C.F., Czerwinski, W., Rucklidge, W.J., 2023. Comprehensive quality assessment of optical satellite imagery using weakly supervised video learning. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Presented at the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Vancouver, BC, Canada, pp. 2125–2135. https://doi.org/10.1109/CVPRW59228.2023.00206.
Paul, S., Ganju, S., 2021. Flood segmentation on Sentinel-1 SAR imagery with semi-supervised learning. https://doi.org/10.48550/arXiv.2107.08369.
Pekel, J.-F., Cottam, A., Gorelick, N., Belward, A.S., 2016. High-resolution mapping of global surface water and its long-term changes. Nature 540, 418–422. https://doi.org/10.1038/nature20584.
Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28.
Saleh, T., Weng, X., Holail, S., Hao, C., Xia, G.-S., 2024. DAM-Net: flood detection from SAR imagery using differential attention metric-based vision transformers. ISPRS J. Photogramm. Remote Sens. 212, 440–453. https://doi.org/10.1016/j.isprsjprs.2024.05.018.
Salur, M.U., Aydın, İ., 2022. A soft voting ensemble learning-based approach for multimodal sentiment analysis. Neural Comput. Appl. 34, 18391–18406. https://doi.org/10.1007/s00521-022-07451-7.
Samela, C., Manfreda, S., Paola, F.D., Giugni, M., Sole, A., Fiorentino, M., 2016. DEM-based approaches for the delineation of flood-prone areas in an ungauged basin in Africa. J. Hydrol. Eng. 21, 06015010. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001272.
Sharma, N., Malik, H., Joshi, R., Saharia, M., 2024. Automated landslide extent estimation from Sentinel 2 data using computationally efficient deep learning. In: IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium. Presented at the IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium, pp. 9579–9581. https://doi.org/10.1109/IGARSS53475.2024.10640552.
Sharma, N., Saharia, M., 2024. ML-CASCADE: a machine learning and cloud computing-based tool for rapid and automated mapping of landslides using earth observation data. Landslides. https://doi.org/10.1007/s10346-024-02360-3.
Sharma, N., Singh, A., Anagha, P., Saharia, M., Dhanya, C.T., 2022. Flood exposure and Social vulnerability during 2020 Assam floods. Authorea Prepr.
Shen, X., Anagnostou, E.N., Allen, G.H., Brakenridge, G.R., Kettner, A.J., 2019a. Near-real-time non-obstructed flood inundation mapping using synthetic aperture radar. Remote Sens. Environ. 221, 302–315.
Shen, X., Wang, D., Mao, K., Anagnostou, E., Hong, Y., 2019b. Inundation extent mapping by synthetic aperture radar: a review. Remote Sens 11, 879.
Sherpa, S.F., Shirzaei, M., 2022. Country-wide flood exposure analysis using Sentinel-1 synthetic aperture radar data: case study of 2019 Iran flood. J. Flood Risk Manag. 15, e12770. https://doi.org/10.1111/jfr3.12770.
Singh, A., Dhanya, C.T., 2024. A novel framework for assessment of human impact of floods: demonstrated for the Indian subcontinent. J. Hydrol. 635, 131110. https://doi.org/10.1016/j.jhydrol.2024.131110.
Singh, A., Gautam, S., Dhanya, C.T., 2024. Increased socio-vulnerability to floods around flood protection structures: case study of Ganga and Brahmaputra basins (India). Hydrol. Sci. J. 69, 2466–2480. https://doi.org/10.1080/02626667.2024.2413014.
Singh, A., Sarma, A.K., Hack, J., 2020. Cost-effective optimization of nature-based Solutions for reducing urban floods considering limited space availability. Environ. Process. 7, 297–319. https://doi.org/10.1007/s40710-019-00420-8.
Singh, K.V., Setia, R., Sahoo, S., Prasad, A., Pateriya, B., 2015. Evaluation of NDWI and MNDWI for assessment of waterlogging by integrating digital elevation model and groundwater level. Geocarto Int. 30, 650–661. https://doi.org/10.1080/10106049.2014.965757.
Tasci, E., Uluturk, C., Ugur, A., 2021. A voting-based ensemble deep learning method focusing on image augmentation and preprocessing variations for tuberculosis detection. Neural Comput. Appl. 33, 15541–15555. https://doi.org/10.1007/s00521-021-06177-2.
Tazmul Islam, M., Meng, Q., 2022. An exploratory study of Sentinel-1 SAR for rapid urban flood mapping on Google Earth Engine. Int. J. Appl. Earth Obs. Geoinformation 113, 103002. https://doi.org/10.1016/j.jag.2022.103002.
Torres, R., Snoeij, P., Geudtner, D., Bibby, D., Davidson, M., Attema, E., Potin, P., Rommen, B., Floury, N., Brown, M., Traver, I.N., Deghaye, P., Duesmann, B., Rosich, B., Miranda, N., Bruno, C., L'Abbate, M., Croci, R., Pietropaolo, A., Huchler, M., Rostan, F., 2012. GMES Sentinel-1 mission. Remote Sens. Environ., The Sentinel Missions - New Opportunities for Science 120, 9–24. https://doi.org/10.1016/j.rse.2011.05.028.
Twele, A., Cao, W., Plank, S., Martinis, S., 2016. Sentinel-1-based flood mapping: a fully automated processing chain. Int. J. Remote Sens.
Uuemaa, E., Ahi, S., Montibeller, B., Muru, M., Kmoch, A., 2020. Vertical accuracy of freely available global digital elevation models (ASTER, AW3D30, MERIT, TanDEM-X, SRTM, and NASADEM). Remote Sens 12, 3482. https://doi.org/10.3390/rs12213482.
Waldner, F., Chen, Y., Lawes, R., Hochman, Z., 2019. Needle in a haystack: mapping rare and infrequent crops using satellite imagery and data balancing methods. Remote Sens. Environ. 233, 111375. https://doi.org/10.1016/j.rse.2019.111375.
Wang, J., Wang, S., Wang, F., Zhou, Y., Wang, Z., Ji, J., Xiong, Y., Zhao, Q., 2022a. FWENet: a deep convolutional neural network for flood water body extraction based on SAR images. Int. J. Digit. Earth 15, 345–361. https://doi.org/10.1080/17538947.2021.1995513.
Wang, L., Li, R., Zhang, C., Fang, S., Duan, C., Meng, X., Atkinson, P.M., 2022b. UNetFormer: a UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. ISPRS J. Photogramm. Remote Sens. 190, 196–214. https://doi.org/10.1016/j.isprsjprs.2022.06.008.
Wang, T., Shi, J., Letu, H., Ma, Y., Li, X., Zheng, Y., 2019. Detection and removal of clouds and associated shadows in satellite imagery based on Simulated radiance fields. J. Geophys. Res. Atmospheres 124, 7207–7225. https://doi.org/10.1029/2018JD029960.
Wen, Z., Zhang, C., Shao, G., Wu, S., Atkinson, P.M., 2021. Ensembles of multiple spectral water indices for improving surface water classification. Int. J. Appl. Earth Obs. Geoinformation 96, 102278. https://doi.org/10.1016/j.jag.2020.102278.
Wieland, M., Fichtner, F., Martinis, S., Groth, S., Krullikowski, C., Plank, S., Motagh, M., 2024. S1S2-Water: a global dataset for semantic segmentation of water bodies from Sentinel- 1 and Sentinel-2 satellite images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 17, 1084–1099. https://doi.org/10.1109/JSTARS.2023.3333969.
Wightman, R., Raw, N., Soare, A., Arora, A., Ha, C., Reich, C., Guan, F., Kaczmarzyk, J., MrT, Mike, SeeFun, Contrastive, Rizin, M., Hyeongchan, Kim, Kertész, C., Dushyant, Mehta, Cucurull, G., Kushajveer Singh, Hankyul, Tatsunami, Y., Lavin, A., Zhuang, Juntang, Hollemans, M., Rashad, M., Sameni, S., Shults, V., Lucain, Wang, X., Kwon, Yonghye, Uchida, Y., 2023. rwightman/pytorch-image-models: v0.8.10dev0 Release. https://doi.org/10.5281/ZENODO.7618837.
Wightman, R., Touvron, H., Jégou, H., 2021. ResNet Strikes Back: an Improved Training Procedure in Timm.
Wortsman, M., Ilharco, G., Gadre, S.Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A.S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., Schmidt, L., 2022. Model Soups: Averaging Weights of Multiple Fine-Tuned Models Improves Accuracy without Increasing Inference Time.
Woznicki, S.A., Baynes, J., Panlasigui, S., Mehaffey, M., Neale, A., 2019. Development of a spatially complete floodplain map of the conterminous United States using random forest. Sci. Total Environ. 647, 942–953. https://doi.org/10.1016/j.scitotenv.2018.07.353.
Xu, H., 2006. Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery. Int. J. Remote Sens. 27, 3025–3033. https://doi.org/10.1080/01431160600589179.

Yan, K., Di Baldassarre, G., Solomatine, D.P., Schumann, G.J.-P., 2015. A review of low-cost space-borne data for flood modelling: topography, flood extent and water level. Hydrol. Process. 29, 3368–3387. https://doi.org/10.1002/hyp.10449.

Yasir, M., Liu, S., Pirasteh, S., Xu, M., Sheng, H., Wan, J., de Figueiredo, F.A.P., Aguilar, F.J., Li, J., 2024. YOLOShipTracker: Tracking ships in SAR images using lightweight YOLOv8. Int. J. Appl. Earth Obs. Geoinformation 134, 104137. https://doi.org/10.1016/j.jag.2024.104137.

Yu, H., Wang, R., Li, P., Zhang, P., 2023. Flood detection in Polarimetric SAR data using Deformable convolutional vision model. Water 15, 4202. https://doi.org/10.3390/w15244202.

Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J., 2020. UNet++: Redesigning skip connections to Exploit Multiscale features in image segmentation. https://doi.org/10.48550/arXiv.1912.05074.