DC4Flood: A deep clustering framework for rapid flood detection using Sentinel-1 SAR imagery

Kasra Rafiezadeh Shahi, *Member, IEEE,* Andrés Camero, Jeremy Eudaric, Heidi Kreibich

***Abstract*—Severe flood losses have been on the rise, and this trend is expected to become increasingly prevalent in the future due to climate and socio-economic changes. Swiftly identifying flooded areas is crucial for mitigating socio-economic losses and facilitating effective recovery. Synthetic Aperture Radar (SAR) sensors are operational in all-weather, day-and-night conditions and offer a rapid, accurate, and cost-effective means of obtaining information for quick flood mapping. However, the complex nature of SAR images, such as speckle noise, coupled with the often absence of training/labeled samples, presents significant challenges in their processing procedures. To alleviate such hindrances, we can benefit from unsupervised classification approaches (also known as clustering). Clustering methods offer valuable insights into newly acquired datasets without the need for training or labeled samples. However, traditional clustering approaches are predominantly linear-based and overlook the spatial information of neighboring pixels during analysis. Thus, to attenuate these challenges, we propose a deep learning (DL)-based clustering approach for flood detection (DC4Flood) using SAR images. The primary advantage of DC4Flood over existing DL-based clustering approaches lies in its ability to capture multi-scale spatial information. This is achieved by utilizing multiple dilated convolutions with varying dilation rates and subsequently fusing the extracted multi-scale information to effectively and efficiently analyze SAR images in an unsupervised manner. Extensive experiments conducted on SAR images from six different flood events demonstrate the effectiveness of the proposed DC4Flood. The code of the work will be available at https://github.com/Kasra2020/DC4Flood.**

***Index Terms*—Deep Learning; Unsupervised Learning; Clustering; Convolutional Autoencoder; Remote Sensing; Sentinel-1; Synthetic Aperture Radar; Flood Detection**

I. Introduction

Effective recovery after natural disasters demands accurate mapping of affected areas [1], [2]. According to [3], natural hazards globally affected $\sim$ 4.2 billion people and caused $\sim$ US$ 2.97 trillion economic losses over the last two decades. Flood losses are severe, and observed trends in the past and projected ones for the future indicate a continuous increase in flood losses caused by climate and socio-economic changes [4]. Hence, it is of great importance to mitigate flood losses through effective emergency management and reconstruction during and shortly after events. For this matter, remote sensing (RS) sensors can play a pivotal role in acquiring fast and precise information on flooded areas. Among different RS sensors, the synthetic aperture radar (SAR) sensor is a versatile tool for obtaining remote sensing data, offering the ability to capture information on Earth's surface dynamics with unprecedented precision. The advantages of SAR include all-weather, day-and-night imaging capabilities, making it an invaluable asset for various Earth monitoring applications such as flood mapping [5].

Over the past two decades, SAR images have proven suitable for flood detection. According to [6], we can divide the traditional processing workflows of SAR images to delineate the inundated areas into four main categories: (1) visual interpretation and digitization, which strongly rely on expert knowledge and can be extensively time-consuming; (2) thresholding of SAR backscatter value, which requires a thorough investigation to determine an optimal threshold; nevertheless, the identified threshold might be subjective and cannot be generic for different regions [7]; (3) image change detection techniques that aim to compare pre- and post-event SAR images [8]; (4) supervised and unsupervised machine learning (ML)-based approaches. Nonetheless, due to the distinct manner in which radar waves interact with terrestrial features compared to optical radiation, it is essential to exercise caution and precision when interpreting radar imagery. In contrast to optical images, radar images are formulated through the coherent interaction between transmitted microwaves and target objects. As a consequence of this coherent interaction, radar imagery suffers from the effects of speckle noise. Such characteristics pose challenges when attempting to directly adapt established optical image analysis approaches to process SAR images [9].

In recent years, deep learning (DL) has prevailed over traditional ML techniques in achieving different tasks, including image classification and signal processing [10], [11]. The increasing maturity of DL-based approaches has led to their widespread adoption across various domains, such as Earth observation [12]. Within this context, SAR image analysis using DL-based approaches has been an active topic, particularly for flood detection. While the majority of existing DL-based approaches for SAR image analysis fall under supervised learning [9], necessitating a substantial amount of training/labeled samples. However, annotating images, particularly in real-time scenarios, poses considerable challenges [13]. In this context, unsupervised classifiers, commonly referred to as clustering methods, offer a valuable avenue to alleviate such obstacles. Despite their proven success and encouraging outcomes in various applications, such as hyperspectral image clustering and LiDAR clustering, there has been relatively limited exploration of clustering approaches, particularly DL-based ones, for detecting inundated areas using SAR im-

K. Rafiezadeh Shahi and H. Kreibich are with Section Hydrology, GFZ German Research Centre for Geosciences, Potsdam, Brandenburg, Germany. A. Camero and J. Eudaric are with German Aerospace Center (DLR), Remote Sensing Technology Institute, EO Data Science, Wessling, Bayern, Germany.
Corresponding e-mail: rafizadehshahie.kasra@gmail.com

agery [7], [8], [14]–[16]. In [7], the authors proposed employing ground range detected (GRD) SAR images, where such images are already preprocessed and made available by the European Space Agency. GRD SAR images, therefore, can be utilized to derive desired information for a region of interest. Their proposed processing workflow contains two main steps. Initially, the workflow starts with making use of classic co-occurrence texture measures integrated with amplitude information within a fuzzy classification framework. This strategy mitigates the critical effect of thresholding. Consequently, a change-detection approach is deployed to the full-resolution GRD SAR image to generate the map of inundated areas. Authors in [15] introduced a weakly supervised DL-based soft clustering for flood detection using SAR imagery. Their proposed approach uses a new assignment strategy named "soft association", where it computes the probability of each pixel belonging to different clusters. Hence, each pixel would be assigned to a cluster with a higher probability value. Apart from their successes in flood detection using SAR imagery, such an approach still relies on training/labeled samples. In a more recent study [8], HuiHui Dong *et al.,* designed a multi-scale DL-based network to integrate clustering and convolutional neural network (CNN) to generate clustering-friendly features from SAR images. The proposed DL-based clustering approach makes use of multi-scale extracted features at different receptive fields. Such an approach ensures incorporating spatial information from the neighboring pixel, which is not located in the closed range. Despite its state-of-the-art (SOTA) performance, the utilization of standard convolutional operations at different scales increases the computational demand, particularly for processing large-scale SAR datasets. This can present limitations in emergency response scenarios where rapid mapping of flooded areas is crucial.

To address this computational constraint, we propose a deep clustering framework that deploys multiple dilated (also known as atrous) convolutions [17] for SAR image analysis in flood detection. This approach enables us to process large-scale SAR datasets efficiently and in an unsupervised fashion. To the best of our knowledge, this is the first attempt to utilize multiple dilated convolutions to extract multi-scale information from SAR images in an unsupervised fashion.

Therefore, the main contributions of this paper are as follows:

- Introducing a deep clustering framework for identifying flood-affected areas through Sentinel-1 SAR imagery. The proposed deep learning-based clustering approach leverages a convolutional autoencoder structure as its backbone to accurately capture the non-linear relationships present among data points and effectively handle the inherent noise in SAR images in an unsupervised fashion.
- While similar approaches have been explored primarily in hyperspectral image clustering, DC4Flood offers a new way to utilize the advantages of SAR imaging for flood mapping.

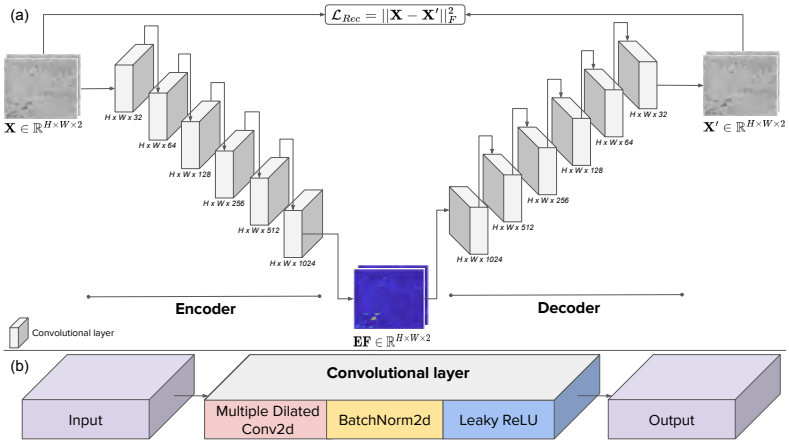

Fig. 1. Illustration of the proposed DC4Flood framework: (a) The architecture of DC4Flood; (b) The deployed convolutional layers in the architecture.

II. Deep Clustering for Flood Detection (DC4Flood)

This section introduces the proposed deep clustering framework for flood detection using SAR images (DC4Flood). In this study, $\mathbf{X} \in \mathbb{R}^{h \times w \times \mathcal{D}}$ and $\mathbf{X'} \in \mathbb{R}^{h \times w \times \mathcal{D}}$ express a SAR and its reconstructed image, respectively, where h and w denote the spatial dimensions (height and width) and $\mathcal{D}$ denotes the number of polarimetric channels. Furthermore, $\mathbf{EF} \in \mathbb{R}^{h \times w \times 2}$ represents the latent features in the bottleneck.

To train a DL-based segmentation model in an unsupervised fashion, we can utilize autoencoder (AE) architectures. The general idea behind AEs is to initially extract informative features from the input data through the encoder phase and then reconstruct the original input data via the decoder phase. The extracted features in the bottleneck are the so-called "latent features", which contain all the essential information to rebuild the original input data. To facilitate unsupervised training, AEs strive to minimize the disparity between the original and reconstructed data. AEs have been successfully adapted from the computer vision field to tackle the existing challenges in geoscience and remote sensing applications [18]. However, the main downfall of standard AEs is that such approaches are pixel-wise, while in many applications, it is shown that incorporating spatial information would improve the classification results [13]. To address this limitation, convolutional AEs (CAEs) are proposed to inject spatial information into the data processing pipeline [17].

In image processing, CAEs have demonstrated their superior performance compared to standard AEs, primarily owing to their incorporation of spatial information [18]. However, existing CAEs typically investigate the spatial information from nearby pixels within close proximity. As a result, this approach overlooks valuable spatial information that may be derived from more distant pixels. To mitigate this challenge, we can extract multi-scale features by increasing the kernel sizes; however, this way can be computationally exhaustive [17]. Alternatively, we can benefit from dilated convolutions (also known as atrous convolutions) to efficiently incorporate the spatial information from different respective fields. Apart from

tremendous endeavors devoted by researchers to tailor the dilated convolutions for different Earth observation applications, yet the capability of such a strategy for analyzing SAR imagery, particularly in the flood detection application has not been fully explored. Thus, in this work, we introduce a DL-based clustering approach (DC4Flood), which benefits from multiple dilated convolutional operations to identify inundated areas using SAR imagery (Fig. 1 (a)). To elucidate the training procedure of our proposed approach, we first formulate a normal convolutional operation as follows:

$$\mathbf{EF} = \mathbf{f}(\sum_{s=1}^{\mathcal{D}} \mathbf{X}_s * \mathbf{W}_i + b_i), \tag{1}$$

where $\mathbf{W}_i$ and b_i represent weight matrix and bias at the $i-th$ convolutional layer, respectively. In addition, $s = 1, 2, ..., \mathcal{D}$ indicates the number of channels. In Eq. 1, $f(.)$ denotes the convolutional operation. However, in Eq. 1, the $f(.)$ merely implies a single-scale convolutional operation. In order to derive multi-scale spatial information, we can deploy the multiple dilated convolutions as:

$$\begin{aligned} \mathbf{EF}_{r1} &= \mathbf{f}_{r1}(\sum_{s=1}^{\mathcal{D}} \mathbf{X}_s * \mathbf{W}_i + b_i) \\ \mathbf{EF}_{r2} &= \mathbf{f}_{r2}(\sum_{s=1}^{\mathcal{D}} \mathbf{X}_s * \mathbf{W}_i + b_i), \end{aligned} \tag{2}$$

Specifically, $\mathbf{f}_{r1}$ corresponds to a standard convolution operation utilizing 3×3 kernels, while $\mathbf{f}_{r2}$ employs 3×3 kernels with a dilation rate of 2. As a result, DC4Flood captures spatial information from both closely positioned pixels and those situated at greater distances. Subsequently, we concatenate the obtained $\mathbf{EF}_{r1}$ and $\mathbf{EF}_{r2}$ feature maps for further analysis. This method assures the incorporation of spatial information across various receptive fields, all while minimizing computational overhead. Therefore, we can express the concatenation and aggregation of extracted multi-scale features as follows:

$$\mathbf{EF}^j = \sigma(\mathrm{Bn}(\mathbf{f}^j(\mathbf{EF}_{r1} \oplus \mathbf{EF}_{r2}))), \tag{3}$$

where $\mathbf{EF}^j$ is the extracted features at the $j-th$ convolutional layer. In Eq. 3, $\oplus$ expresses the concatenation operation of multi-scale features, and $\mathbf{f}^j$ denotes 1×1 convolutional filters to aggregate the concatenated features in the last step. As shown in Fig. 1 (b), we apply the batch normalization (Bn) process as the next step to have computational consistency across different convolutional layers. Consequently, an activation function (σ) is deployed to capture the non-linear intrinsic relation between the data points in SAR imagery (in this study, we utilized the Leaky Rectified Linear unit (LeakyReLU)).

In this study, we deploy the mean squared error (MSE) as the loss function to train the network in an unsupervised fashion. The aim, hence, is to minimize the loss value and optimize the network parameters (i.e., weights and bias) as follows:

$$\arg\min_{\mathbf{W},\mathbf{b}} \mathcal{L}_{Rec} = ||\mathbf{X} - \mathbf{X}'||_F^2, \tag{4}$$

where $||.||_F$ denotes the *Frobenius*-norm. Once the training process is complete, meaning the specified number of iterations has been reached, we use $\mathbf{EF} \in \mathbb{R}^{h \times w \times 2}$ in the bottleneck for the clustering purpose. As the last step, to generate the final clustering map, *k*-means clustering is applied on $\mathbf{EF}$ [13].

III. Experimental results and discussion

A. Sen1floods11 SAR data

To validate the performance of DC4Flood, we used SAR images from six flood events sourced from the Sen1floods11 benchmark dataset. The Sen1flood11 dataset comprises 11 flood events in different ecosystems. Each scene in the dataset is derived from Sentinel-1 level-1 ground range detected (GRD) product and contains two polarimetric channels (i.e., VV, VH) with the spatial dimension of 512×512 pixels. For our experiments, we chose six flood events (Bolivia, Ghana, India, Mekong, Spain, and the USA), for which labeled data was meticulously generated by trained remote sensing analysts. The selection of these events was deliberate, aiming to represent diverse flood occurrences, ecosystems, and continents, thereby ensuring the robustness and reliability of our analysis.

B. Experimental implementation details

All the models are implemented in Python, version 3.9.6. The DL-based approaches are implemented using PyTorch 2.0.1 on the high-performance computing (HPC) facility at GFZ. We used Adam as the model optimizer for DL-based approaches. As suggested by many studies in this field, the Adam optimizer is configured with default parameter values, with $\beta 1 = 0.9$, $\beta 2 = 0.999$, $\epsilon = 10^{-8}$, weight decay= 0. In contrast, the learning rate is determined as 0.01 based on a trial-and-error process. We configured the number of iterations to 200 for all DL clustering approaches, ensuring stable performance. Additionally, we utilized the default hyperparameters of the *k*-means algorithm implemented in Scikit-learn 1.4.1 for clustering. The implementation of DC4Flood will be available online at: https://github.com/Kasra2020/DC4Flood.

C. Evaluation metrics

In this work, we utilized the four following evaluation metrics:

- Overall accuracy (OA): is the widely used supervised evaluation metric, where OA reports how many pixels are correctly labeled [18].
- Normalized mutual information (NMI): is an established unsupervised evaluation metric, which provides the reduction in entropy of cluster labels when the cluster labels are given [19].
- Adjusted rand index (ARI): is another well-known unsupervised evaluation metric. ARI reports the similarity between two partitions (i.e., clustering result and ground truth data) [19].
- Silhouette coefficient (SiC): is a fully unsupervised evaluation metric that does not require ground truth data. SiC indicates how well each pixel is clustered with respect to its assigned cluster [20].

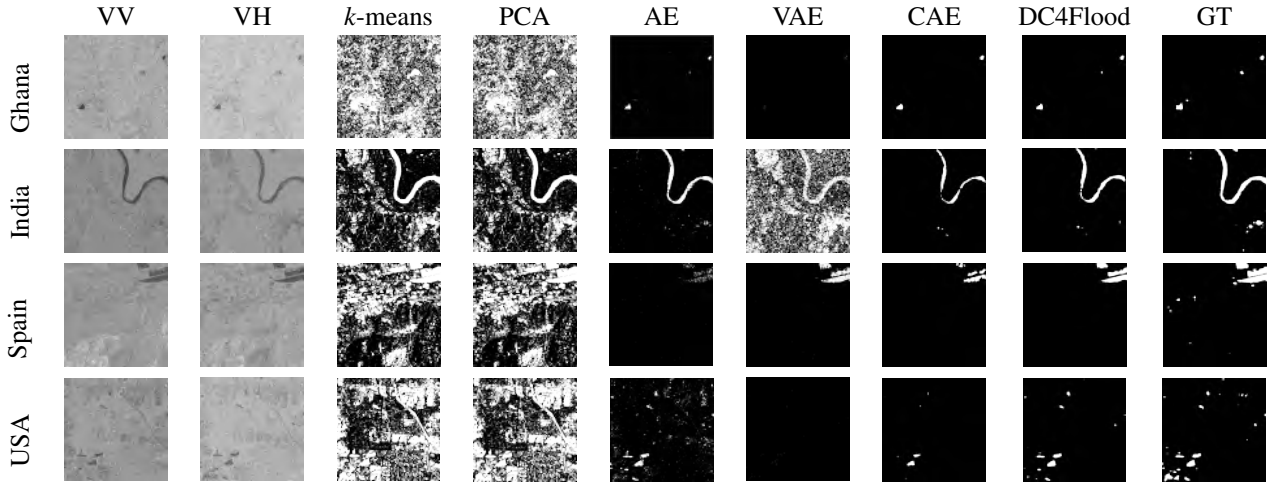

Fig. 2. The Ghana, India, and USA SAR images, their corresponding ground truth data, and their respective generated clustering maps using *k*-means, PCA, AE, VAE, CAE, DC4Flood. Please note that the color scheme, white and black present the flooded and non-flooded regions, respectively.

D. Clustering performance

- Training process of DC4Flood: Table.II displays the optimization requirements for DC4Flood, varying with different convolutional operations (standard or dilated). Increasing the kernel window size leads to a substantial rise in parameters $(53,260,020)$, while utilizing dilated convolutions significantly reduces them $(30,844,660)$, ensuring effective capture of multi-scale spatial information with lower computational load. We explored dilation rates of 1, 2, and 4, but dilation rate 4 did not improve results, possibly due to SAR image noise structure. Nevertheless, depending on the applications, additional hyperparameter tuning may be required. Furthermore, Fig. 3 shows the convergence of DC4Flood after 200 iterations across all six test sites. Notably, stable performance is achieved after 100 iterations, suggesting that reducing the iteration count from 200 to 100 would not significantly impact DC4Flood's performance.
- Comparison to SOTA clustering approaches: We compared our proposed algorithm with five different clustering approaches to detect flooded areas, where two are classical approaches (i.e., *k*-means [21] and principal component analysis (PCA) [22]) and two DL-based ones (i.e., AE [18], variational AE (VAE) [23], CAE [18]). In order to have a fair comparison across DL-based approaches, we set a similar number of extracted features at each layer as 32, 64, 128, 256, 512, and 1024. Moreover, for the convolutional-based clustering approaches (i.e., CAE and DC4Flood), the kernel size has been set to 3×3. The quantitative assessment of clustering approaches, as an average over five independent runs, is in Table I. The results show that DL-based clustering approaches can prevail over classical clustering approaches. As reported in Table I, there are improvements in OA by deploying DL-based of about $20-39$ points in Bolivia, Ghana, Spain, and USA datasets, while the improvement is only $1-30$ points for the India and Mekong datasets. In accordance with the overall result, we can observe that incorporating spatial information within the training process leads to higher accuracy in terms of different evaluation metrics. DC4Flood attains the highest OA compared to all studied clustering approaches in different case study areas. Nevertheless, DC4Flood and CAE have relatively similar performances across different areas. Such an observation might be due to the nature of SAR images acquired in these regions.

Fig. 2 illustrates the original SAR polarimetric channels (i.e., VV and VH) for different case study areas, their corresponding ground truth, and generated clustering maps. At first glance, one can observe that the classical models (i.e., *k*-means and PCA) produce the so-called "noisy" labels. In other words, such classical approaches tend to generate heterogeneous patches of clusters. These behaviors can be explained by (1) their linear nature, which prevents them from capturing the non-linear relation between pixels in a SAR image; (2) they do not explore the spatial information from the neighboring pixels. As depicted in Fig. 2, DL-based approaches generate more homogeneous patches of clusters; nonetheless, detailed examination reveals that the generated clustering maps by both AE and VAE still suffer from the "salt-pepper" issue, while by incorporating spatial information, we observe that this issue is significantly mitigated. As stated before, CAE and DC4Flood have relatively similar results, but the detailed examination of generated clustering maps unveils that the results produced by DC4Flood agree more with the ground truth data in different regions. This observation indicates that including multi-scale spatial information within the clustering process can improve the delineation of inundated areas using SAR images.

IV. Conclusion

This article introduces a deep clustering framework, denoted as DC4Flood, designed for the identification of flooded regions through SAR imagery. Employing an unsupervised training approach, DC4Flood leverages a CAE structure to effectively

TABLE I
THE QUANTITATIVE COMPARISON OF STUDIED CLUSTERING APPROACHES IN TERMS OF OA, NMI, ARI, AND SIC. THE RESULTS ARE REPORTED AS AN AVERAGE OF 5 INDEPENDENT RUNS.

Country	Evaluation metric	Classic		Deep learning			
		k-means	PCA	AE	VAE	CAE	DC4Flood
Bolivia	OA(%)	60.78 ± 0.00	60.96 ± 0.00	81.76 ± 0.16	82.25 ± 0.19	93.75 ± 0.12	**96.99 ± 0.03**
	NMI(%)	0.70 ± 0.00	0.71 ± 0.00	2.41 ± 0.02	1.94 ± 0.02	26.87 ± 0.20	**30.22 ± 0.20**
	ARI(%)	0.28 ± 0.00	0.28 ± 0.00	3.93 ± 0.06	3.84 ±0.04	30.37 ± 0.21	**37.95 ± 0.22**
	SiC(%)	38.53 ± 0.00	38.56 ± 0.00	77.82 ± 0.20	49.92 ± 0.20	89.09 ± 0.16	**89.14 ± 0.06**
Ghana	OA(%)	60.22 ± 0.00	60.40 ± 0.00	86.74 ± 0.20	82.27 ± 0.24	96.66 ± 0.06	**99.86 ± 0.00**
	NMI(%)	1.14 ± 0.00	1.15 ± 0.00	36.99 ± 0.33	12.44 ± 0.08	46.87 ± 0.33	**72.82 ± 0.03**
	ARI(%)	0.44 ± 0.00	0.45 ± 0.00	44.63 ± 0.41	10.62 ± 0.10	55.68 ± 0.36	**82.21 ± 0.03**
	SiC(%)	35.65 ± 0.00	35.68 ± 0.00	83.56 ± 0.16	63.07 ± 0.29	92.69 ± 0.11	**99.34 ± 0.00**
India	OA(%)	85.11 ± 0.00	85.10 ± 0.00	86.09 ± 0.18	83.97 ± 0.17	91.74 ± 0.11	**97.65 ± 0.01**
	NMI(%)	20.45 ± 0.00	20.44 ± 0.00	**38.48 ± 0.33**	20.69 ± 0.06	28.23 ± 0.33	37.86 ± 0.31
	ARI(%)	23.34 ± 0.00	23.32 ± 0.00	45.40 ± 0.40	20.35 ± 0.06	34.15 ± 0.40	**45.94 ± 0.38**
	SiC(%)	51.05 ± 0.00	51.05 ± 0.00	86.29 ± 0.12	66.99 ± 0.23	89.01 ± 0.14	**98.37 ± 0.01**
Mekong	OA(%)	64.31 ± 0.00	64.40 ± 0.00	69.09 ± 0.16	77.61 ± 0.13	70.28 ± 0.15	**94.55 ± 0.20**
	NMI(%)	10.73 ± 0.00	10.76 ± 0.00	22.48 ± 0.01	20.09 ± 0.06	29.26 ± 0.11	**33.18 ± 0.10**
	ARI(%)	2.98 ± 0.04	4.02 ± 0.04	29.78 ± 0.07	22.98 ± 0.02	32.80 ± 0.12	**36.57 ± 0.09**
	SiC(%)	42.42 ± 0.00	42.42 ± 0.00	75.77 ± 0.17	45.19 ± 0.17	76.75 ± 0.16	**91.38 ± 0.10**
Spain	OA(%)	74.69 ± 0.00	74.86 ± 0.00	93.25 ± 0.05	89.45 ± 0.19	94.97 ± 0.06	**98.25 ± 0.01**
	NMI(%)	10.46 ± 0.00	10.53 ± 0.00	14.83 ± 0.04	27.28 ± 0.22	22.20 ± 0.31	**45.80 ± 0.27**
	ARI(%)	9.14 ± 0.00	9.26 ± 0.00	17.13 ± 0.04	35.44 ± 0.29	30.75 ± 0.31	**56.33 ± 0.27**
	SiC(%)	42.24 ± 0.00	42.31 ± 0.00	89.71 ± 0.08	62.49 ± 0.22	91.40 ± 0.14	**96.75 ± 0.05**
USA	OA(%)	64.19 ± 0.00	63.98 ± 0.00	81.00 ± 0.24	82.63 ± 0.22	92.11 ± 0.15	**98.86 ± 0.00**
	NMI(%)	4.01 ± 0.00	3.98 ± 0.00	19.25 ± 0.17	20.08 ± 0.05	34.23 ± 0.21	**37.72 ± 0.22**
	ARI(%)	2.21 ± 0.00	2.17 ± 0.00	29.20 ± 0.28	28.92 ± 0.05	43.43 ± 0.28	**48.47 ± 0.28**
	SiC(%)	41.36 ± 0.00	41.32 ± 0.00	85.16 ± 0.12	72.24 ± 0.22	89.82 ± 0.12	**97.79 ± 0.01**

TABLE II
THE NUMBER OF LEARNABLE PARAMETERS FOR DC4FLOOD USING DIFFERENT CONVOLUTIONAL OPERATIONS.

Approach	No. parameters
DC4Flood (standard conv)	53,260,020
DC4Flood (dilated conv)	30,844,660

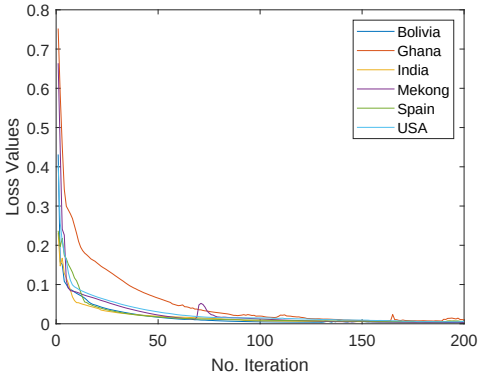

Fig. 3. Illustration of the convergence of DC4Flood across all test sites.

deal with the inherent noise present in SAR images. Furthermore, the model utilizes multiple dilated convolutions with different dilation rates, enabling it to exploit spatial information from adjacent pixels across various receptive fields. Experimental results underscore the superior performance of DC4Flood when compared to other studied clustering approaches. As a result, such an unsupervised framework can be a valuable tool for decision-makers to promptly respond to flood situations.

ACKNOWLEDGMENT

This work is funded by the Helmholtz AI project SURF ((semi/un)supervised machine learning flood damage assessment) grant ZT-I-PF-5-125.

REFERENCES

[1] F. Cian, M. Marconcini, and P. Ceccato, "Normalized difference flood index for rapid flood mapping: Taking advantage of EO big data," *Remote Sensing of Environment*, vol. 209, pp. 712–730, 2018.

[2] Y. A. Buehler, T. W. Kellenberger, D. Small, and K. I. Itten, "Rapid mapping with remote sensing data during flooding 2005 in Switzerland by object-based methods: A case study," *WIT Transactions on Ecology and the Environment*, vol. 89, 2006.

[3] N. Yaghmaei, *Human Cost of Disasters: An Overview of the Last 20 Years, 2000-2019*. UN Office for Disaster Risk Reduction, 2020.

[4] D. Paprotny, A. Sebastian, O. Morales-Nápoles, and S. N. Jonkman, "Trends in flood losses in Europe over the past 150 years," *Nature communications*, vol. 9, no. 1, p. 1985, 2018.

[5] C. W. Tay, S.-H. Yun, S. T. Chin, A. Bhardwaj, J. Jung, and E. M. Hill, "Rapid flood and damage mapping using synthetic aperture radar in response to Typhoon Hagibis, Japan," *Scientific data*, vol. 7, no. 1, p. 100, 2020.

[6] R. Manavalan, "SAR image analysis techniques for flood area mapping-literature survey," *Earth Science Informatics*, vol. 10, no. 1, pp. 1–14, 2017.

[7] D. Amitrano, G. Di Martino, A. Iodice, D. Riccio, and G. Ruello, "Unsupervised rapid flood mapping using Sentinel-1 GRD SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 6, pp. 3290–3299, 2018.

[8] H. Dong, W. Ma, L. Jiao, F. Liu, and L. Li, "A multiscale self-attention deep clustering for change detection in SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[9] X. X. Zhu, S. Montazeri, M. Ali, Y. Hua, Y. Wang, L. Mou, Y. Shi, F. Xu, and R. Bamler, "Deep learning meets SAR: Concepts, models, pitfalls, and perspectives," *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, no. 4, pp. 143–172, 2021.

[10] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[12] C. Persello, J. D. Wegner, R. Hänsch, D. Tuia, P. Ghamisi, M. Koeva, and G. Camps-Valls, "Deep learning and earth observation to support the sustainable development goals: Current approaches, open challenges, and future opportunities," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 2, pp. 172–200, 2022.

[13] K. R. Shahi, P. Ghamisi, B. Rasti, R. Gloaguen, and P. Scheunders, "Ms^{2}a-net: Multiscale spectral–spatial association network for hyperspectral image clustering," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 6518–6530, 2022.

[14] K. R. Shahi, P. Ghamisi, B. Rasti, P. Scheunders, and R. Gloaguen, "Unsupervised data fusion with deeper perspective: A novel multisensor deep clustering algorithm," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 284–296, 2022.

[15] R. Yadav, A. Nascetti, H. Azizpour, and Y. Ban, "Unsupervised flood detection on SAR time series using variational autoencoder," *International Journal of Applied Earth Observation and Geoinformation*, vol. 126, p. 103635, 2024.

[16] H. Zhai, H. Zhang, P. Li, and L. Zhang, "Hyperspectral image clustering: Current achievements and future lines," *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, no. 4, pp. 35–67, 2021.

[17] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[18] B. Rasti, D. Hong, R. Hang, P. Ghamisi, X. Kang, J. Chanussot, and J. A. Benediktsson, "Feature extraction for hyperspectral imagery: The evolution from shallow to deep: Overview and toolbox," *IEEE Geoscience and Remote Sensing Magazine*, vol. 8, no. 4, pp. 60–88, 2020.

[19] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: is a correction for chance necessary?" in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 1073–1080.

[20] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[21] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[22] A. Maćkiewicz and W. Ratajczak, "Principal components analysis (PCA)," *Computers & Geosciences*, vol. 19, no. 3, pp. 303–342, 1993.

[23] A. Tasissa, D. Nguyen, and J. Murphy, "Deep diffusion processes for active learning of hyperspectral images," *arXiv preprint arXiv:2101.03197*, 2021.