

An Interpretable Deep Semantic Segmentation Method for Earth Observation

1st Ziyang Zhang

*School of Computing and Communications
Lancaster University
Lancaster, UK
z.zhang51@lancaster.ac.uk*

2nd Plamen Angelov

*School of Computing and Communications
Lancaster University
Lancaster, UK
p.angelov@lancaster.ac.uk*

3rd Eduardo Soares

*School of Computing and Communications
Lancaster University
Lancaster, UK
e.almeidasoares@lancaster.ac.uk*

4rd Nicolas Longepe

*Phi-Lab Explore Office
European Space Agency
Frascati, Italy
nicolas.lungepe@esa.int*

5rd Pierre Philippe Mathieu

*Phi-Lab Explore Office
European Space Agency
Frascati, Italy
pierre.philippe.mathieu@esa.int*

Abstract—Earth observation is fundamental for a range of human activities including flood response as it offers vital information to decision makers. Semantic segmentation plays a key role in mapping the raw hyper-spectral data coming from the satellites into a human understandable form assigning class labels to each pixel. Traditionally, water index based methods have been used for detecting water pixels. More recently, deep learning techniques such as U-Net started to gain attention offering significantly higher accuracy. However, the latter are hard to interpret by humans and use dozens of millions of abstract parameters that are not directly related to the physical nature of the problem being modelled. They are also labelled data and computational power hungry. At the same time, data transmission capability on small nanosatellites is limited in terms of power and bandwidth yet constellations of such small, nanosatellites are preferable, because they reduce the revisit time in disaster areas from days to hours. Therefore, being able to achieve as highly accurate models as deep learning (e.g. U-Net) or even more, to surpass them in terms of accuracy, but without the need to rely on huge amounts of labelled training data, computational power, abstract coefficients offers potentially game-changing capabilities for EO (Earth observation) and flood detection, in particular. In this paper, we introduce a prototype-based interpretable deep semantic segmentation (IDSS) method, which is highly accurate as well as interpretable. Its parameters are in orders of magnitude less than the number of parameters used by deep networks such as U-Net and are clearly interpretable by humans. The proposed here IDSS offers a transparent structure that allows users to inspect and audit the algorithm’s decision. Results have demonstrated that IDSS could surpass other algorithms, including U-Net, in terms of IoU (Intersection over Union) total water and Recall total water. We used WorldFloods data set for our experiments and plan to use the semantic segmentation results combined with masks for permanent water to detect flood events.

Index Terms—Earth observation, semantic segmentation, flood detection, interpretable deep learning, prototype-based classifiers, U-Net, WorldFloods

I. INTRODUCTION

Flood is one of the most catastrophic weather events as it caused about 7 million fatalities in the twentieth century [1]. The average annual loss generated by floods are estimated at over US\$ 100 billion (2015) [2]. Recent findings claim that the exposure to floods is expected to grow three times by 2050 due to increases in population and economic assets in flood-prone areas [3]. Depending on the socio-economic scenario, human losses from flooding are projected to rise by 70–83% and direct flood damage by 160–240% relative to 1976–2005 [4]. Detecting flooding and its associated impacts is critical to effective risk reduction [5], [6].

The advent of Copernicus of ESA (European Space Agency) and the launch of several Sentinels satellites have provided massive amounts of data for a range of Earth observation missions, including flood detection [7]. SAR (Synthetic Aperture Radar) images and optical remote sensing images are often used for flood mapping, but the optical remote sensing images is preferable due to its mature processing and analysis techniques, while SAR images suffer from noise and information loss problems [8]. The Sentinel-2 satellite constellation is often used for flood detection due to the availability of Multi-Spectral Imagers, shorter revisit times (5 days), and higher spatial resolution (10m for some bands) [9].

Flood mapping traditionally relies on water indices, such as NDWI [10] and MNDWI [11]. However, these methods are threshold-based, which requires expert knowledge. In recent years, machine learning algorithms have also been applied to flood detection. [12] uses clustering algorithm and a combination of several water indexes to identify the water areas. However, the coastal water, clouds, shadows and snow pixels had to be removed in advance, which greatly increases the workload.

The recent work from [13] investigates how a constellation of small nano satellites assembled from commercial off-the-

The project is co-funded by the ESA (European Space Agency).

978-1-6654-5656-2/22/\$31.00 ©2022 IEEE | DOI: 10.1109/IS57118.2022.10019621

shelf (COTS) hardware, also known as CubeSats, could be used for disaster response in the case of floods. The authors proposed the use of deep learning algorithms to produce multi-class segmentation with high accuracy on-board of a very cheap satellite hardware. Although the deep learning approaches proposed for this challenging task have produced great results in terms of accuracy, they are black-box and are extremely difficult to audit [14].

In this paper, a new prototype-based approach is proposed called interpretable deep semantic segmentation (IDSS), which extends the recently introduced explainable Deep Neural Network (xDNN) [15] through a new clustering and decision mechanism. The prototype-based nature of IDSS allows clear interpretability of its decision mechanism [16]. This is important for the human-in-the-loop process involved in this application. Results demonstrate that IDSS is able to surpass xDNN and U-Net in terms of IoU and recall for water detection.

II. LITERATURE REVIEW

A. Water/Flood mapping

Water/Flood mapping requires semantic segmentation for allocating a class label to each multidimensional pixel. Traditionally, different water index-based methods are being used to determine liquid water. They represent ratios of different spectral bands from the raw satellite signal that can characterize the water absorption. The rationale behind the usefulness of the water indices is that the water absorbs energy in the near infrared (NIR) and short-wave infrared (SWIR) wavelengths, making it significantly different from other objects on the ground [17]. The most widely used water index is NDWI [10]. Other water indices also exists such as MNDWI [11], WNDWI [18] and AWEI [19]. However, such methods often require manual setting of thresholds, which is challenging and the choice does influence the result significantly.

Machine learning methods were also applied for flood mapping. For example, K-means was used to perform clustering based on Synthetic Aperture Radar (SAR) optical features and thresholds were applied to further perform classification by [20]. The SVM classifier was used as a reference label since it is considered to provide better results than the thresholding methods [21]. KNN was used to perform per-pixel classification based on the water index by [22].

More recently, with the rapid development of deep learning, convolutional neural networks started to be used more widely for Earth Observation and flood mapping, in particular. For example, the fully convolutional neural network Resnet50 trained on Sentinel-1 satellite images was used to segment permanent water and flood by [23], U-Net and a simple CNN were trained on Sentinel-2 satellite images used to perform the onboard flood segmentation task by [13]. The main advantage of using deep convolutional networks is their high levels of accuracy measured primarily by IoU (intersection over union) and recall characteristics for this particular problem. They also offer powerful latent features extraction capability. However, the downside is that they require large amount of

labeled training data and computational power and most of all, they lack interpretability. They are often considered as "black box" models because they have millions of abstract model parameters (weights) which have no direct physical meaning and are hard to interpret or check if affected by noise or adversarial actions especially when transmitted. Attempts have been made to provide some explainability, but these are mostly *post-hoc* partial solutions or surrogate models. Therefore, currently, there is a powerful trend aiming to develop explainable or interpretable-by-design alternatives that are as powerful as such Deep Neural Networks, yet offer human-intelligible models [24], [15], [14].

The method proposed in this paper offers visual and linguistic forms of interpretation based on prototypes, offering layers with clear interpretation as well as a linguistic *IF... THEN* rules and clear decision making process based on similarity. These forms of representation of the model can be inspected by a human and have clear meaning. The prototypes can also be visualized using RGB colours as well as raw features such as NIR, SWIR, etc. which are easy to interpret by a human.

In this next sub-section, prototype-based machine learning methods are briefly reviewed because they are underestimated and often overlooked, but they do offer clear advantages in terms of interpretability and high levels of accuracy and performance.

B. Prototype-based models

Prototype-based machine learning methods have similar concepts as some methods from cognitive psychology and neuroscience in regards to comparisons of new observations or stimuli to a set of prototypes [25]. Prototype-based machine learning models have been attracting much attention due to their easy to understand decision making processes and interpretability. K nearest neighbor [26] and K-means [27] are the most representative algorithms based on prototypes. The most typical prototype-based neural network algorithm is the Radial-basis Function (RBF) method [28] which is like a bridge between neural networks and linguistic *IF...THEN* rules. There are also other prototype-based machine learning algorithms such as fuzzy c-means clustering [29] and Learning Vector Quantization (LVQ) [30].

Recently, with the rapid development of deep learning algorithms, prototype-based models have a tendency to integrate with neural networks. Such as ProtoPNet [31], xDNN [15] and a nonparametric segmentation framework [32]. The IDSS method proposed in this paper differs from these works because xDNN [15] does not perform clustering to generate prototypes and has not been applied to flood mapping while the work by [32] only look for prototypes in the embedding space, which greatly reduces the interpretability of the model. The method proposed in this paper not only outperforms the state-of-the-art deep convolutional networks such as UNet and SCNN as well as various water-indices-based models, but also, greatly improves the interpretability of the model by further finding the mean value in the raw features space corresponding

to the prototypes (cluster centers) in the latent features space and can be described by linguistic *IF... THEN* rules.

III. PROPOSED METHOD

In this paper a new method is proposed named interpretable deep semantic segmentation (IDSS). It is prototype-based unlike the majority of the alternatives which are highly parameterized. IDSS considers that each class can be divided into different sub-classes in the latent feature space. These can result from clustering the data of a given class (e.g. water) or by selecting peaks of the data distribution as in the xDNN approach [15]. Clustering can be done in the raw or latent features space. IDSS is using clustering. In this paper the mini-batch K-means [33] clustering method is used on the latent features space without loss of generality. The correspondence between the latent feature space representation and the raw feature space representation for the cluster centers is being preserved by performing the mini-batch K-means algorithm on the latent feature space, but performing all averaging (mean) operations to both, the raw and the latent feature space representations. In this way, all the decisions about reallocations of points to clusters are made based on the latent feature representation; however, each center is being exactly represented in both, the raw and latent feature space. The raw feature space representation is more suitable to produce an interpretable linguistic *IF...THEN* form of the prototypes/means. This semantic form of representation facilitates the human perception and analysis of the decisions made by the algorithm.

The training architecture of the proposed IDSS method is shown in Fig. 1. It is described in more detail below:

1) Raw features layer: The Raw features layer is responsible for extracting each pixel, x from the image and assigning it to different classes using corresponding labels since our algorithm is trained per class. In this layer the raw features (e.g. the 13 features from the hyper-spectral signal of the Sentinel-2 satellite images as described in *Table I*) are extracted per pixel. This vector of raw features is further passed to the next later which may be clustering or (optionally) - feature transformation.

TABLE I: Hyper-spectral 13-band signal from the Sentinel-2 satellite (vector of raw features)

Name	Description	Resolution (m)
B01	Coastal aerosol	60
B02	Blue	10
B03	Green	10
B04	Red	10
B05	Vegetation red edge	20
B06	Vegetation red edge	20
B07	Vegetation red edge	20
B08	NIR	10
B8A	Narrow NIR	20
B09	Water vapour	60
B10	SWIR	60
B11	SWIR	20
B12	SWIR	20

2) Latent features layer (optional): This layer is optional. For a better performance, the raw features can be transformed, e.g. by orthogonalization and optimization using PCA [34]. Alternatively, deep feature representations using pre-trained convolutional neural network such as Segnet [35] or U-Net [36] can be used. It is also possible to directly feed raw features into the next clustering layer. In this paper, U-Net Deep Neural Network that is pre-trained on Worldfloods dataset [13] is used to perform feature extraction. The input to the U-Net is formed by the raw 13 bands hyper-spectral Sentinel-2 pixel values from the training images and the penultimate layer of the U-Net is used for feature extraction. The output is, thus, a 64-dimensional feature vector.

For all pixels, x_j^i ($j = 1, 2, \dots, N$; where N denotes the number of pixels of a given image) for each class i (where $i = 1, 2, \dots, c$), the U-Net is used to map the raw features vector x_j^i into a n -dimensional latent feature space ϕ , where each feature is represented as F_j^i . In this paper, $F_j^i \in R^{64}$, i.e. $n = 64$, since the output of the penultimate layer of the U-Net is a 64-dimensional vector. Then, L2 normalization is performed to the (latent) features as shown below:

$$F_j^i = \frac{F_j^i}{\|F_j^i\|} \quad (1)$$

For $i = 1, 2, \dots, c, j = 1, 2, \dots, N$.

3) Clustering layer: The Clustering layer is responsible for two main tasks: applying clustering to the latent features and finding the cluster centers, C_m^i where m represents the number of cluster centers per class. Each cluster center does have a corresponding representation in the raw features (x_j^i) domain which can serve as a prototype providing a clear interpretability.

The mini-batch K-means [33] clustering method is used in this paper without lose of generality. This particular version of the K-means approach is reported to be more suitable for handling large data sets and can be updated online. The main output of the K-means algorithm is the set of cluster centers, C_m^i which minimize the distance between clusters in terms of the latent features used, $F_j^i \in F$.

The choice of the number of centers m is important for the performance of the mini-batch K-means algorithm. In this paper, m is set to 500, which means that each of the land, water and clouds classes have 500 corresponding prototypes.

4) Prototypes layer: The prototypes are considered to correspond to the identified centres, C_m^i ; however, they are being expressed in terms of raw features. Since the use of latent features is optional (in particular, one of the results from this study shown in Table III is using raw features and no latent ones), prototypes may directly correspond to the centers. In cases, when latent features are being used, a transformation is needed which is a result of finding the means at each step of the k-means algorithm in terms of both, raw and latent features. It has to be stressed, however, that the decision how to reallocate data samples between clusters which is a key part of the K-means approach is based on latent features (if they are

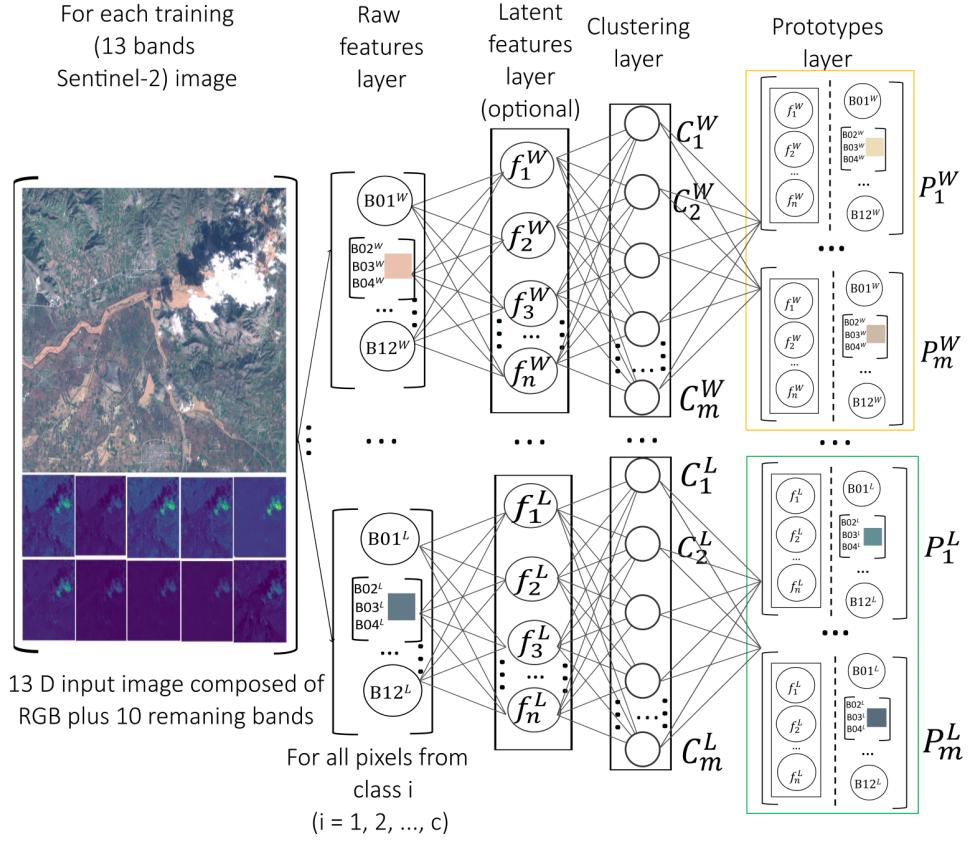


Fig. 1: Training process of IDSS

being used). The raw features are convenient to communicate the result to the human users. In particular, the prototypes can generate linguistic *IF ... THEN* rules, as shown in Fig. 3, where \sim denotes "is similar to" and "Coastal", "SWIR", etc., represent the features (e.g. the 13 bands of the Sentinel-2 signal) for a single pixel. Such linguistic rules are easy to understand, explain and use by humans because they are intuitive. They can be applied to unlabeled test images as a tool to understand the decision making process for deciding which class a particular pixel is more likely to belong to. One such linguistic rule can be formed per prototype connecting individual terms such as (SWIR ...), (Coastal ...) etc. with logical conjunction (AND). In particular, each term concerns an individual raw feature (SWIR, Coastal,...). Brought together with the logical conjunction, AND operator they form an *IF...THEN* rule where the *THEN* condition defines the class label (e.g. Water). In this paper, 500 clusters are considered per class and, therefore, 500 such *IF...THEN* rules are being generated per class. All these 500 rules can be combined together using logical disjunction, OR operator resulting in a single (but large) linguistic rule per class. Although such rule is large it is quite clear and easy to interpret, understand and explain to a human. In a summary, the outcome can be either one large linguistic rule per class or multiple smaller rules per class.

The training procedure of IDSS is summarised below:

The validation architecture of IDSS is responsible for making predictions on new unlabeled Sentinel-2 test images, as shown in Fig. 2, described in detail below:

1) Raw features layer: This layer is responsible for extracting each pixel x from the unlabeled Sentinel-2 test image quite similarly to the similar layer of the Training architecture.

2) Latent features layer (optional): This layer is the same as the training architecture, 64-dimensional feature value is extracted from the penultimate layer of the U-Net. then, L2 normalization is performed same as in the training architecture.

3) Prototypes layer: This layer is responsible for calculating the similarity to each prototype from each class. Similarity, S is calculated in the latent F_j (or raw, x_j) feature space using exponential kernel:

$$S(F^*, P_j^i) = \exp(-\|F^* - P_j^i\|^2) \quad (2)$$

where F^* denotes a pixel from the test (unlabeled, new) image represented in the form of the latent feature and P denotes the j^{th} prototypes of the i^{th} class.

4) Decision-making layer: The decision-making layer is responsible for assigning labels to unknown pixels. In this study, K nearest neighbours, also called the "few winners take all" method, is used to make the final decision. In this study, K is set to 10, which means that the labels corresponding to the ten prototypes with the highest similarity to the unlabeled

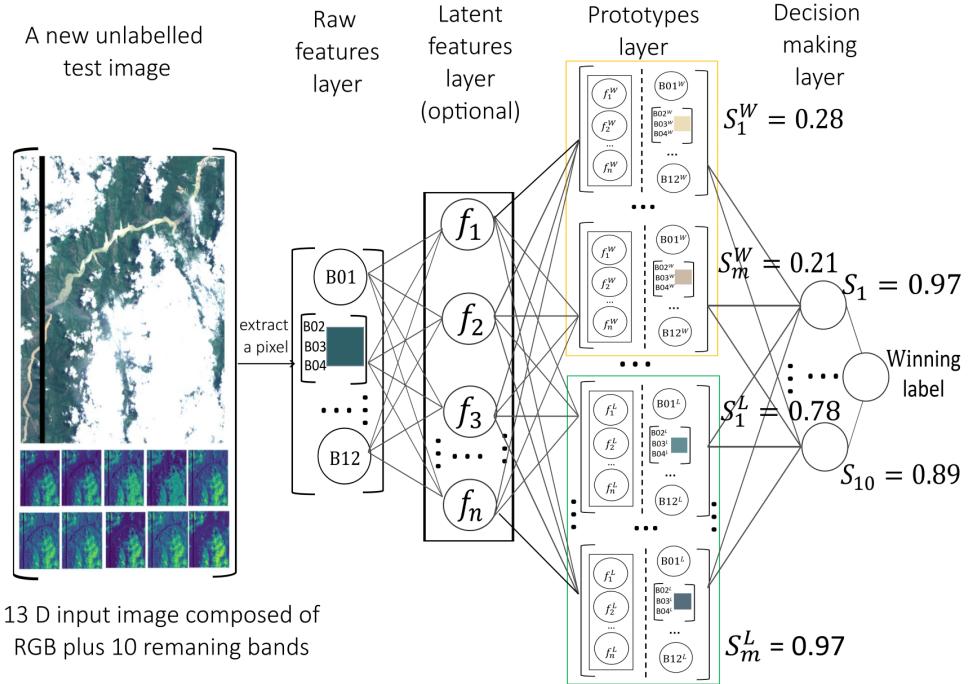


Fig. 2: Validation process of IDSS

```

IF ((Coastal aerosol ~ 0.1480) AND (  ~  ) AND ... AND (SWIR ~ 0.0218)) OR ...
... OR ((Coastal aerosol ~ 0.1394) AND (  ~  ) AND ... AND (SWIR ~ 0.0292))
THEN (Water)

IF ((Coastal aerosol ~ 0.1453) AND (  ~  ) AND ... AND (SWIR ~ 0.0805)) OR ...
... OR ((Coastal aerosol ~ 0.1252) AND (  ~  ) AND ... AND (SWIR ~ 0.0636))
THEN (Land)

```

Fig. 3: Linguistic rules of IDSS

pixels are used to perform decision-making. The label can be obtained by the formula (3).

$$L(F^*) = \arg \max_{\beta \in c} \sum_{i=1}^K \delta(\beta, l(F_i)) \quad (3)$$

where L denotes the winning label, $l(F_i)$ denotes the label of k nearest neighbors of F^* , $\delta(\beta, l(F_i)) = 1$ if $\beta = l(F_i)$ and $\delta(\beta, l(F_i)) = 0$ otherwise.

IV. EXPERIMENT SETUP

A. Worldfloods Dataset

In this paper to test the newly proposed IDSS method the rich and recent Worldfloods dataset [13] is used. It describes 119 real flood events and has 424 flood maps. All flood maps are based on Sentinel-2 images containing 13 bands which form the raw features vector. The description of each channel

is shown in Table I [37] [38] and a statistical analysis of the data is shown in Table II [13]. The bands with a resolution larger than 10m were downsampled to 10m using nearest neighbours interpolation. From Table II it can be observed that the percentage of could pixels in the training data set is much higher than that in the validation and test data sets. Such an anomaly also takes place for the land pixel from the validation data set and the flood water pixels from the test data set. The Worldfloods data set is highly imbalanced (with a very small proportion of water pixels), which is a serious challenge for the classification task.

B. Training

The implementation of the proposed model is based on the ML4Floods pipeline [13] and scikit-learn [39]. The U-Net has been pre-trained on the Worldfloods dataset and used for extraction of the latent features. The Sentinel-2 images from the training data set were cropped to size 256×256 for feature extraction via U-Net. Different from the convolutional neural network, the proposed IDSS does not require a validation data set to check the model performance. In this paper the 6 validation Sentinel-2 images were used as a training data set which also demonstrates the ability of the proposed method to work with a smaller labelled training data sets.

C. Testing

The test images are cut into 256×256 for feature extraction. It should be noted that Worldfloods images vary in size significantly. Most of the images cannot be cut entirely to the size of 256×256 . The patches that are with a smaller size than 256×256 are discarded in training, but this can not be

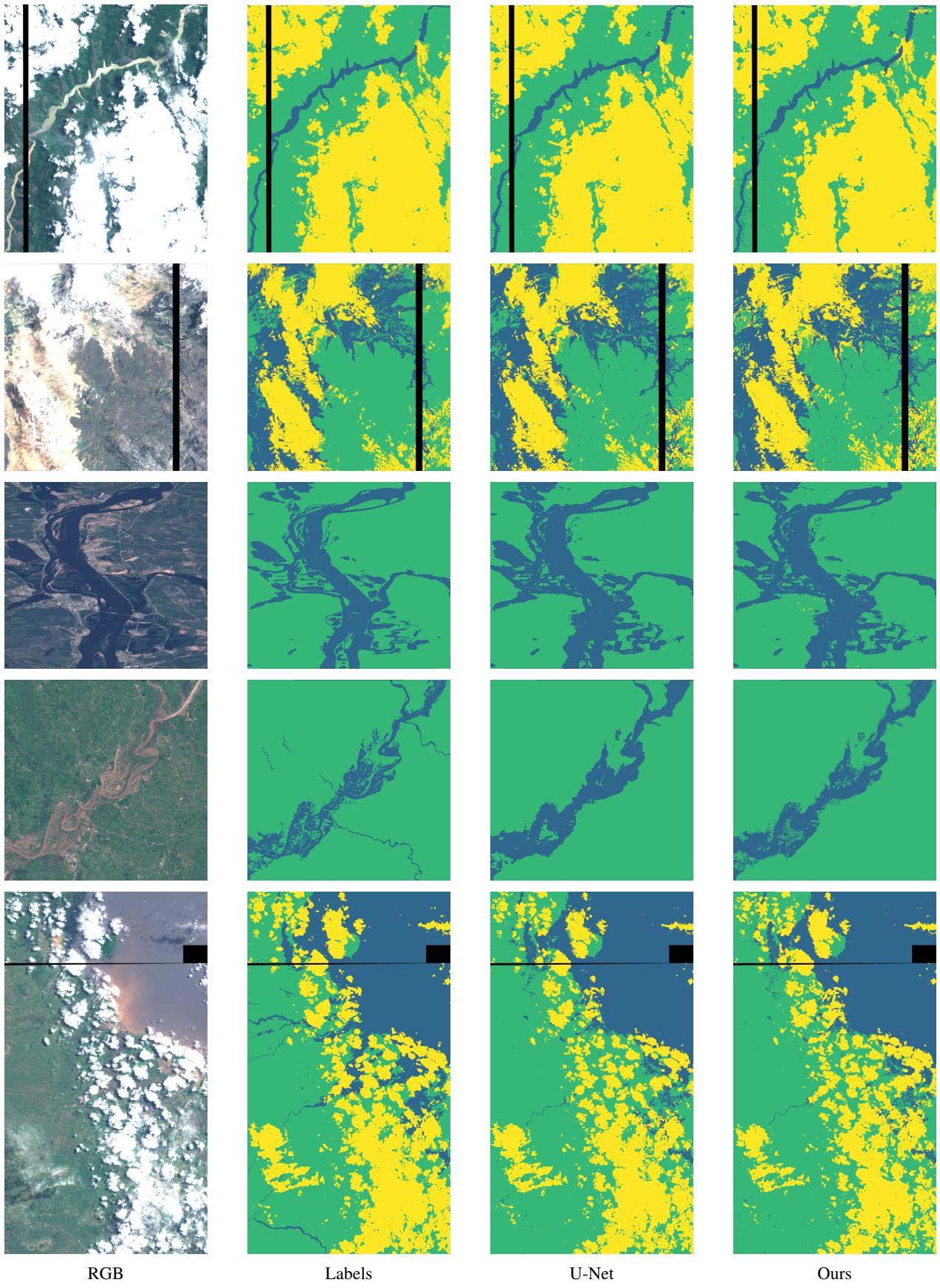


Fig. 4: Comparison of segmentation results. The meaning of the colours is shown below: green - land, yellow - cloud, blue - water.

TABLE II: Statistics of the Worldfloods dataset

Dataset	<i>Flood events</i>	<i>Flood maps</i>	<i>Water pixels (%)</i>		<i>Land pixels</i>	<i>Cloud pixels</i>	<i>Invalid pixels</i>
			Flood	Permanent	(%)	(%)	(%)
Training	108	407	1.45	1.25	43.24	50.25	3.81
Validation	6	6	3.14	5.19	76.72	13.27	1.68
Test	5	11	20.23	1.16	59.05	16.21	3.34

done during testing. In this paper padding is performed on the test images so that they can be perfectly cut into 256×256 patches. After that, the prediction masks are stuck together, the padding part is simply being cut off. All experiments in this paper were conducted on the High-End Computing Cluster with Intel(R) Xeon(R) Gold 6248 CPU @ 2.50 GHz processor.

D. Performance Evaluation

The following metrics were used to evaluate the proposed algorithm:

Intersection over Union (IoU):

$$IoU = \frac{TP}{TP + FP + FN} \quad (4)$$

Recall:

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

Here, TP, FP and FN represent True Positive, False Positive and False Negative, respectively. IoU is the widely used measure of success for this problem because it maximizes the TP while minimizing FP as well as TP and FN. Recall is considered to be the more important measure than Precision since missing flood areas are more dangerous and risky than over-predicting flood areas.

V. RESULTS

The results of the proposed IDSS method are presented in Table III and summarized in this section.

TABLE III: Comparison of IoU and recall results

Model	<i>IoU</i> total water %	<i>Recall</i> total water %	Parameters (x1000)
<i>IDSS¹</i> (ours)	73.10	93.35	96
<i>IDSS²</i> (ours)	70.03	96.11	96
xDNN [15]	71.50	90.27	20
U-Net [13]	72.42	95.42	7790
SCNN [13]	71.12	94.09	260
<i>NDWI¹</i> [10]	65.12	95.75	-
Linear [13]	64.87	95.55	-
<i>NDWI²</i> [10]	39.99	44.84	-

Table III shows the metric comparison between the proposed algorithms and a number of state-of-the-art methods, including threshold-based and Deep Neural Network algorithms. It can be observed that the proposed algorithm achieves very competitive results going beyond the state-of-the-art. While trained only on six images, the proposed method *IDSS¹* has 0.68% IoU total water improvement compared to U-Net. At the same time, the proposed IDSS method also offers linguistic form of explainability (Fig. 3) and a clear network architecture

(Fig. 1 and Fig. 2) and is using in orders of magnitude less model parameters. *IDSS²* trained on 206 training images has 0.36% Recall total water improvement compared to the second highest method *NDWI¹*. The threshold used for *NDWI¹* was -0.22 and for *NDWI²* it was 0 .

xDNN is an interesting recently proposed explainable form of Deep Neural Network method which in this paper was trained on the raw Sentinel-2 images without feature extraction and, again, only trained on six images. In this paper before applying this method the Worldfloods model was used to remove the clouds. the result using xDNN after cloud removal is also competitive with the IoU total water of 71.50%. More importantly, however, the xDNN algorithm provides direct interpretability through the linguistic *IF ... THEN* rules and the decision making can be understood and analyzed by humans, which can play an important role in practical applications.

A visual comparison of the segmentation results is shown in Fig. 4. It can be appreciated visually that the segmentation results obtained by the proposed method are very similar to those obtained by the U-Net model and to the reference labels. U-Net model tends to over-predict water areas if compare the images in the fourth row.

VI. CONCLUSION

In this paper, an interpretable deep semantic segmentation (IDSS) method is proposed for Earth Observation. It provides human interpretable results while ensuring a performance on par or surpassing that of the mainstream and state-of-the-art convolutional neural network. The proposed IDSS is a prototype-based method. Prototypes are the local focal points that are most representative in terms of the features. Working in a latent feature space a projection to the raw feature space allows a human interpretable linguistic *IF...THEN* rules to be used to explain the decision making involved in determining the winning class label. Semantic rules are interpretable and easy for humans to understand allowing users could inspect and audit the decision making process. Worldfloods dataset was used, and the results show that IDSS outperforms the other methods in terms of IoU total water and Recall total water requiring in orders of magnitude less parameters and training data. The future work will focus on investigation of the application of the model to flood detection.

REFERENCES

- [1] B. Merz, G. Blöschl, S. Vorogushyn, F. Dottori, J. C. Aerts, P. Bates, M. Bertola, M. Kemter, H. Kreibich, U. Lall *et al.*, “Causes, impacts and patterns of disastrous river floods,” *Nature Reviews Earth & Environment*, vol. 2, no. 9, pp. 592–609, 2021.

- [2] C. Wannous and G. Velasquez, "United nations office for disaster risk reduction (unisdr)—unisdr's contribution to science and technology for disaster risk reduction and the role of the international consortium on landslides (icl)," in *Workshop on World Landslide Forum*. Springer, 2017, pp. 109–115.
- [3] B. Jongman, P. J. Ward, and J. C. Aerts, "Global exposure to river and coastal flooding: Long term trends and changes," *Global Environmental Change*, vol. 22, no. 4, pp. 823–835, 2012.
- [4] F. Dottori, W. Szewczyk, J.-C. Ciscar, F. Zhao, L. Alfieri, Y. Hirabayashi, A. Bianchi, I. Mongelli, K. Frieler, R. A. Betts *et al.*, "Increased human and economic losses from river flooding with anthropogenic warming," *Nature Climate Change*, vol. 8, no. 9, pp. 781–786, 2018.
- [5] K. Birmingham, J. Fielding, and D. Thrush, "it'll never happen to me': understanding public awareness of local flood risk," *Disasters*, vol. 32, no. 2, pp. 216–238, 2008.
- [6] S. N. Willner, C. Otto, and A. Levermann, "Global economic response to river floods," *Nature Climate Change*, vol. 8, no. 7, pp. 594–598, 2018.
- [7] M. Onoda and O. R. Young, *Satellite earth observations and their impact on society and policy*. Springer Nature, 2017.
- [8] Q. Zhang, P. Zhang, and X. Hu, "Unsupervised grnn flood mapping approach combined with uncertainty analysis using bi-temporal sentinel-2 msr imageries," *International Journal of Digital Earth*, vol. 14, no. 11, pp. 1561–1581, 2021.
- [9] C. Cavallo, M. N. Papa, M. Gargiulo, G. Palau-Salvador, P. Vezza, and G. Ruello, "Continuous monitoring of the flooding dynamics in the albufera wetland (spain) by landsat-8 and sentinel-2 datasets," *Remote Sensing*, vol. 13, no. 17, p. 3525, 2021.
- [10] S. K. McFeeters, "The use of the normalized difference water index (ndwi) in the delineation of open water features," *International journal of remote sensing*, vol. 17, no. 7, pp. 1425–1432, 1996.
- [11] H. Xu, "Modification of normalised difference water index (ndwi) to enhance open water features in remotely sensed imagery," *International journal of remote sensing*, vol. 27, no. 14, pp. 3025–3033, 2006.
- [12] M. C. Cordeiro, J.-M. Martinez, and S. Peña-Luque, "Automatic water detection from multidimensional hierarchical clustering for sentinel-2 images and a comparison with level 2a processors," *Remote Sensing of Environment*, vol. 253, p. 112209, 2021.
- [13] G. Mateo-Garcia, J. Veitch-Michaelis, L. Smith, S. V. Oprea, G. Schumann, Y. Gal, A. G. Baydin, and D. Backes, "Towards global flood mapping onboard low cost satellites with machine learning," *Scientific reports*, vol. 11, no. 1, pp. 1–12, 2021.
- [14] P. P. Angelov, E. A. Soares, R. Jiang, N. I. Arnold, and P. M. Atkinson, "Explainable artificial intelligence: an analytical review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 5, p. e1424, 2021.
- [15] P. Angelov and E. Soares, "Towards explainable deep neural networks (xdnn)," *Neural Networks*, vol. 130, pp. 185–194, 2020.
- [16] J. Bien and R. Tibshirani, "Prototype selection for interpretable classification," *The Annals of Applied Statistics*, vol. 5, no. 4, pp. 2403–2424, 2011.
- [17] L. Ji, L. Zhang, and B. Wylie, "Analysis of dynamic thresholds for the normalized difference water index," *Photogrammetric Engineering & Remote Sensing*, vol. 75, no. 11, pp. 1307–1317, 2009.
- [18] Q. Guo, R. Pu, J. Li, and J. Cheng, "A weighted normalized difference water index for water extraction using landsat imagery," *International journal of remote sensing*, vol. 38, no. 19, pp. 5430–5445, 2017.
- [19] G. L. Feyisa, H. Meilby, R. Fensholt, and S. R. Proud, "Automated water extraction index: A new technique for surface water mapping using landsat imagery," *Remote Sensing of Environment*, vol. 140, pp. 23–35, 2014.
- [20] L. Landuyt, N. E. Verhoest, and F. M. Van Coillie, "Flood mapping in vegetated areas using an unsupervised clustering approach on sentinel-1 and-2 imagery," *Remote Sensing*, vol. 12, no. 21, p. 3611, 2020.
- [21] A. Sekertekin, "A survey on global thresholding methods for mapping open water body using sentinel-2 satellite imagery and normalized difference water index," *Archives of Computational Methods in Engineering*, vol. 28, no. 3, pp. 1335–1347, 2021.
- [22] F. Pan, X. Xi, and C. Wang, "A comparative study of water indices and image classification algorithms for mapping inland surface water bodies using landsat imagery," *Remote Sensing*, vol. 12, no. 10, p. 1611, 2020.
- [23] D. Bonafilia, B. Tellman, T. Anderson, and E. Issenberg, "Sen1floods11: A georeferenced dataset to train and test deep learning flood algorithms for sentinel-1," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 210–211.
- [24] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.
- [25] M. Biehl, B. Hammer, and T. Villmann, "Prototype-based models in machine learning," *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 7, no. 2, pp. 92–111, 2016.
- [26] E. Fix and J. L. Hodges, "Discriminatory analysis. nonparametric discrimination: Consistency properties," *International Statistical Review/Revue Internationale de Statistique*, vol. 57, no. 3, pp. 238–247, 1989.
- [27] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.
- [28] M. D. Buhmann, "Radial basis functions," *Acta numerica*, vol. 9, pp. 1–38, 2000.
- [29] J. C. Bezdek, R. Ehrlich, and W. Full, "Fcml: The fuzzy c-means clustering algorithm," *Computers & geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.
- [30] T. Kohonen, "Learning vector quantization," in *Self-organizing maps*. Springer, 1995, pp. 175–189.
- [31] C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su, "This looks like that: deep learning for interpretable image recognition," *Advances in neural information processing systems*, vol. 32, 2019.
- [32] T. Zhou, W. Wang, E. Konukoglu, and L. Van Gool, "Rethinking semantic segmentation: A prototype view," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2582–2593.
- [33] D. Sculley, "Web-scale k-means clustering," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 1177–1178.
- [34] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.
- [35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [36] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [37] M. Drusch, U. Del Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort *et al.*, "Sentinel-2: Esa's optical high-resolution mission for gmes operational services," *Remote sensing of Environment*, vol. 120, pp. 25–36, 2012.
- [38] D. Phiri, M. Simwanda, S. Salekin, V. R. Nyirenda, Y. Murayama, and M. Ranagalage, "Sentinel-2 data for land cover/use mapping: a review," *Remote Sensing*, vol. 12, no. 14, p. 2291, 2020.
- [39] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.