

Comparative analysis of activation functions in U-Net for binary water segmentation using Sentinel-2 imagery

Pavlo Kundenko[†], Viktoria Hnatushenko[†], Vladyslav Tsaryk[†] and Iryna Dmytriieva[†]

Ukrainian State University of Science and Technologies, 2 Lazariana St, 49005, Dnipro, Ukraine

Abstract

The study examines how different activation functions influence the performance of a U-Net model applied to binary water-body segmentation in Sentinel-2 imagery. Using an identical training setup for each experiment, six nonlinearities—ReLU, Leaky ReLU, ELU, PReLU, Swish and RReLU—are individually substituted into the network while all other parameters remain fixed. Comparative evaluation on a held-out validation set reveals that Leaky ReLU provides the most balanced trade-off between precision and recall, making it the preferred choice for accurate water-mask generation. PReLU offers a similar but slightly lower performance, whereas ELU excels at capturing additional water pixels at the cost of more false positives. The findings highlight the importance of activation-function selection in remote-sensing segmentation tasks and suggest further exploration of advanced nonlinearities and larger, more diverse datasets to enhance generalization.

Keywords

Water-body segmentation, Sentinel-2, U-Net, activation functions, remote sensing, deep learning

1. Introduction

Monitoring water resources, including rivers, lakes, and coastal areas, plays a crucial role in modern research related to sustainable environmental management, agrotechnology, and ecology. Thanks to an extensive satellite network, particularly the Sentinel-2 program, it is now possible to obtain high-resolution multispectral imagery for the regular assessment of water bodies. However, the task of automatically and accurately separating water from land (segmentation) remains challenging due to factors such as water turbidity, seasonal variability, cloud coverage, and the spectral similarity of various landscape elements.

Traditional algorithms based on indices such as the Normalized Difference Water Index (NDWI) offer fast solutions, but they are often vulnerable to complex environmental conditions. With advances in deep learning within the field of computer vision, there is a growing trend towards the use of deep convolutional neural networks (CNNs), which enable more accurate identification of image features [1,2]. One of the most widely used models for segmentation tasks is the U-Net, proposed by Ronneberger et al. Its encoder-decoder architecture with skip connections enables the fusion of deep semantic information with high spatial resolution.

Nonetheless, the effectiveness of CNN training – including that of U-Net – depends not only on architectural design but also on the choice of activation functions, which define how neurons respond

ISW-2025: Intelligent Systems Workshop at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025), May 15–16, 2025, Kharkiv, Ukraine

* Corresponding author.

[†] These authors contributed equally.

✉ pavel.kundenko@gmail.com (P. Kundenko); vvitagnat@gmail.com (Vik. Hnatushenko); tsarick.vlad@gmail.com (V. Tsaryk); irina_dmitrieva@ukr.net (I. Dmytriieva)

📄 0009-0000-9388-2783 (P. Kundenko); 0000-0001-5304-4144 (Vik. Hnatushenko); 0000-0002-9375-4121 (V. Tsaryk); 0009-0008-3298-7563 (I. Dmytriieva)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

to incoming signals. The ReLU (Rectified Linear Unit) family remains the most used due to its simplicity and immunity to the vanishing gradient problem in the positive domain. However, numerous modifications of ReLU (e.g., Leaky ReLU, PReLU, RReLU) as well as alternative functions (ELU, Swish, Mish) have been proposed to improve convergence and address the limitations of standard ReLU.

This study presents a comparative analysis of six activation functions (ReLU, Leaky ReLU, ELU, PReLU, Swish, RReLU) in the context of binary segmentation of water bodies using Sentinel-2 satellite imagery. To ensure objective evaluation, all training parameters (number of epochs, dataset) were fixed so that the only changing variable was the activation function. The evaluation criteria included F1 score, Intersection over Union (IoU), precision, recall, and convergence rate. The results provide insights into which activation function contributes most effectively to accurate and reliable water segmentation under diverse landscapes and imaging conditions.

2. Methodology

U-Net Architecture

This study employs a modified U-Net architecture specifically designed for the binary segmentation of water bodies based on Sentinel-2 satellite data. The choice of U-Net is justified by its ability to integrate high-level (global) features with fine-grained local details—an essential quality for detecting small water bodies and complex shoreline structures. To enhance performance, the input data undergoes preprocessing, including the generation of training patches from satellite scenes and the creation of corresponding target masks. This enables the model to operate effectively on multispectral imagery by adjusting the number of input channels and scale according to the available spectral bands.

During model development, the unique characteristics of Sentinel-2 imagery are considered such as the varying spatial resolution of individual bands and uneven surface illumination [23]. Each image is normalized and divided into fixed-size patches, which simplifies the training process and reduces the need to store large full-resolution intermediate results. Each patch is input into the U-Net as a tensor, typically with 3 or 4 channels (RGB or RGB plus near-infrared). In this project, 512×512 sized patches are used, striking a balance between computational cost and spatial detail preservation.

The architecture retains the classic encoder-decoder structure with convolutional and transposed convolutional blocks. The encoder progressively reduces spatial resolution while extracting increasingly abstract features, whereas the decoder reconstructs the original image dimensions and focuses on the accurate localization of segmented objects. Skip connections between corresponding levels of the encoder and decoder help retain crucial fine-grained information that would otherwise be lost—essential for delineating the boundaries of water bodies [6]. The final output layer generates a binary map, where each pixel is assigned a probability of belonging to the water class. For full-size imagery, the model processes each patch individually and subsequently reassembles the outputs into a single map using mosaicking. Post-processing smoothing techniques are applied to reduce potential artifacts or misclassifications.

The advantages of U-Net in this project are further supported by multiple studies demonstrating its effectiveness for remote sensing data [5]. The model's scalable filter size and layer dimensions allow it to perform robustly under various imaging conditions while preserving the ability to identify narrow linear structures. In this work, the encoder comprises layers with 64, 128, 256, and 512 filters, while the bottleneck block reaches 1024 filters—a configuration widely used for high-resolution image segmentation tasks [9, 10, 3]. To improve the model's sensitivity to water, a near-infrared channel (Sentinel-2 B8) may be incorporated alongside standard RGB, as this band provides better water-land contrast due to differences in spectral reflectance [11]. Ultimately, each 512×512 patch is processed

independently, and the results are aggregated into a continuous segmentation map, optimizing computation while accounting for spatial variability across the image.

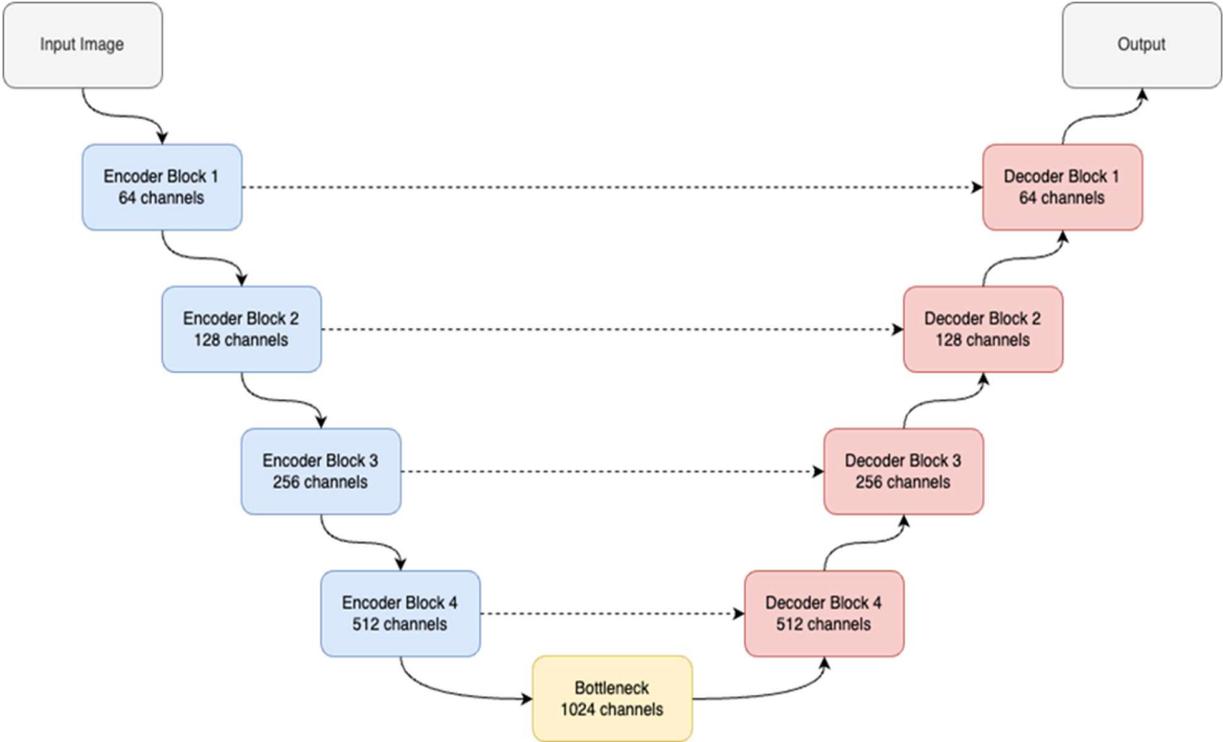


Figure 1: Structure of U-net model used in research

Model Learning and Evaluation

The research begins with the selection of Sentinel-2 satellite image that contains the relevant spectral bands for distinguishing between water and land. The image is normalized to minimize differences in illumination and acquisition conditions.



Figure 2: Selected Sentinel-2 image (ID: S2A_MSIL2A_20241023T090031_N0511_R007_T36UUA_20241023T122451.SAFE)

The fragment (5376x5376 pixels) of the original satellite image was selected for training dataset. The image is then divided into fixed-size patches of 512x512 pixels to simplify the training process and optimize computational efficiency. The total number of patches used for training is 110.

Next, binary masks are generated to indicate the presence of water at the pixel level. Some of these masks are refined manually, while others are derived from spectral indices and later validated for labeling errors. This approach enables the creation of a robust training and validation dataset with a balanced representation of water and non-water regions. A baseline U-Net model is used for segmentation, with the number of input channels tailored to the selected spectral bands. Batch size, learning rate, and other hyperparameters remain constant across all experiments to ensure fair comparison among activation functions.



Figure 3: Image fragment used for neural network training

The encoder-decoder structure, along with max-pooling and transposed convolution operations, allows the network to preserve local details while reconstructing spatial features of the input image.

For each activation function under consideration (ReLU, Leaky ReLU, ELU, PReLU, Swish, RReLU), a separate variant of the model is implemented in which only the activation layers are modified. Aside from the activation function, all other components—including the dataset and training duration—remain unchanged.

Upon completion of training, the models are evaluated on a validation set using F1, IoU, precision, recall, and convergence metrics.

The predicted outputs are stitched into complete segmentation maps, enabling both visual and quantitative assessment of water body detection quality. Finally, a comparative analysis of all six activation functions is performed to identify the most effective one for binary water segmentation from Sentinel-2 imagery.

3. Experiments

ReLU Activation Function

The Rectified Linear Unit (ReLU) is one of the most used activation functions in modern deep convolutional networks [4]. It operates by zeroing all negative input values while retaining a linear relationship for positive inputs. It is defined as:

$$f(x) = \max(0, x) \quad (1)$$

ReLU was introduced to mitigate the vanishing gradient problem often encountered with sigmoid or tanh activations [7]. Unlike these nonlinearities, ReLU provides a constant gradient for positive inputs and avoids costly exponential computations, resulting in faster training.

Its main advantages include computational simplicity and the ability to maintain non-zero gradients when $x > 0$, which facilitates effective optimization in deep architectures [12]. Additionally, the lack of saturation for positive inputs allows neurons to output arbitrarily large values, assuming supportive data and weights. However, ReLU has a significant drawback in the form of "dead neurons"—units that output zero across all inputs if they remain in the negative region during training [8]. Despite this, ReLU continues to demonstrate reliable performance in high-resolution image segmentation, including satellite imagery [13].

In this study, ReLU serves as the baseline activation function. The U-Net model with ReLU is used as a reference to evaluate the performance gains achieved by its alternatives (Leaky ReLU, ELU, etc.). It remains a widely adopted standard due to its proven efficacy in segmentation, classification, and various deep learning tasks [4, 7, 12]. An example of model execution is presented below:

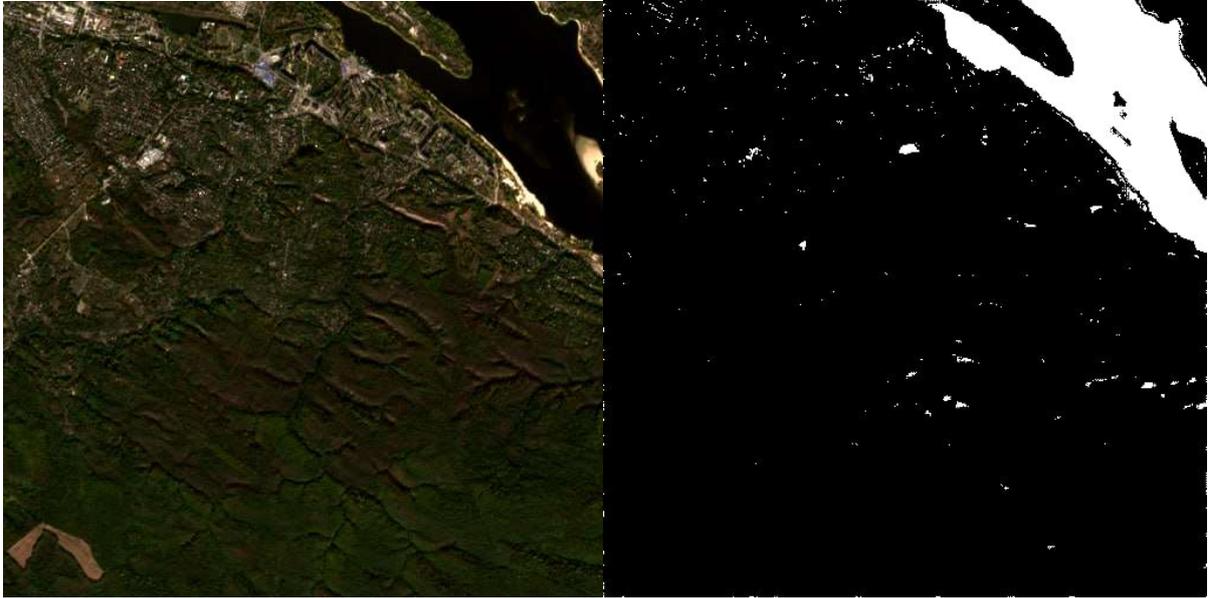


Figure 4: Example of UNetReLU water segmentation result

Leaky ReLU

Leaky ReLU is a variant of ReLU designed to mitigate the issue of "dead neurons" by allowing a small, non-zero gradient for negative input values. While ReLU completely discards negative signals, Leaky ReLU applies a small slope α to retain some gradient information [8]. It is defined as:

$$f(x) = \max(\alpha x, x) \quad (2)$$

where α is a small positive coefficient (e.g., 0.01). This modification reduces the risk of permanent neuron inactivity during training [14].

The performance of Leaky ReLU is sensitive to the choice of α . A very small value makes it behave like ReLU, whereas a large value may weaken its ability to discriminate between signal polarities [15]. In practice, α is often chosen between 0.01 and 0.1 to balance learning speed and neuron activity, especially in vision tasks and satellite image segmentation [16].

In water segmentation tasks, Leaky ReLU may enhance model adaptability in regions with high spectral variability, such as vegetated shorelines or partially flooded zones. This study examines whether Leaky ReLU enables the network to retain more informative neurons and achieve superior segmentation performance compared to baseline ReLU [10].

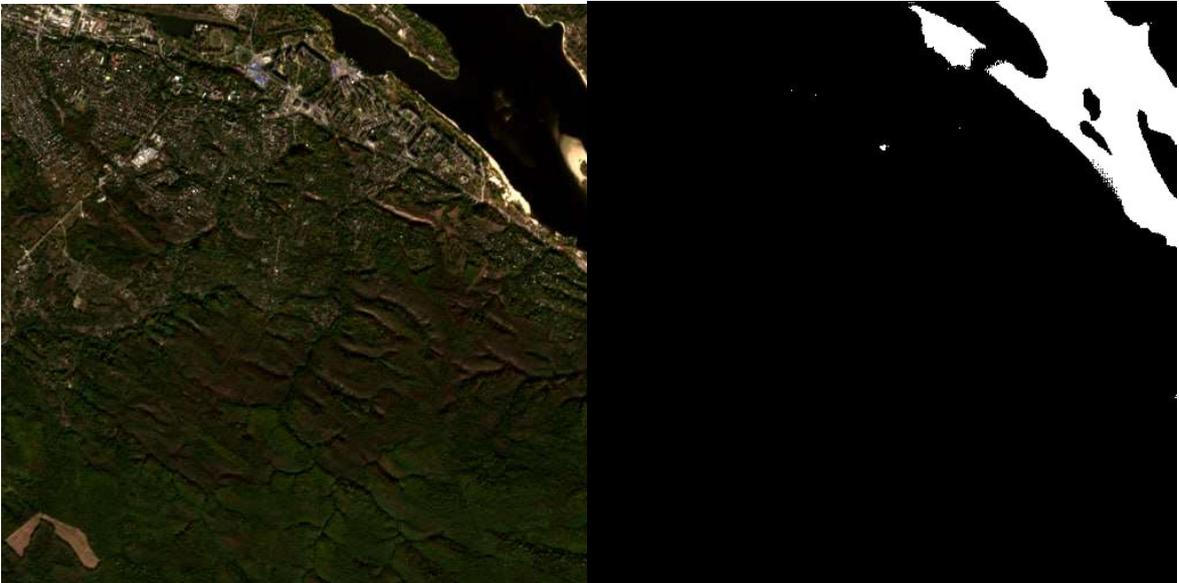


Figure 5: Example of UNet Leaky ReLU water segmentation result

ELU

The Exponential Linear Unit (ELU) was introduced to accelerate convergence and reduce bias shift during training. It is defined as:

$$f(x) = \begin{cases} x, & \text{for } x \geq 0 \\ \alpha(e^x - x), & \text{for } x < 0 \end{cases} \quad (3)$$

where α is a positive parameter, typically set to 1 [17]. Unlike ReLU, ELU produces smooth negative outputs rather than hard zeros, preserving gradients in the negative region [14]. For $\mathbf{x} \geq \mathbf{0}$, it behaves similarly to ReLU, ensuring simple optimization and avoiding saturation [4].

ELU also helps to center activation values around zero, which can facilitate learning and reduce reliance on normalization techniques [18]. However, it incurs higher computational costs due to the exponential term and may generate large negative outputs that destabilize training in some cases [19].

In this study, ELU is evaluated in contexts where nuanced control over negative inputs is beneficial—for instance, near noisy land-water transitions. The goal is to assess whether ELU can accelerate learning and improve segmentation metrics compared to ReLU and Leaky ReLU.

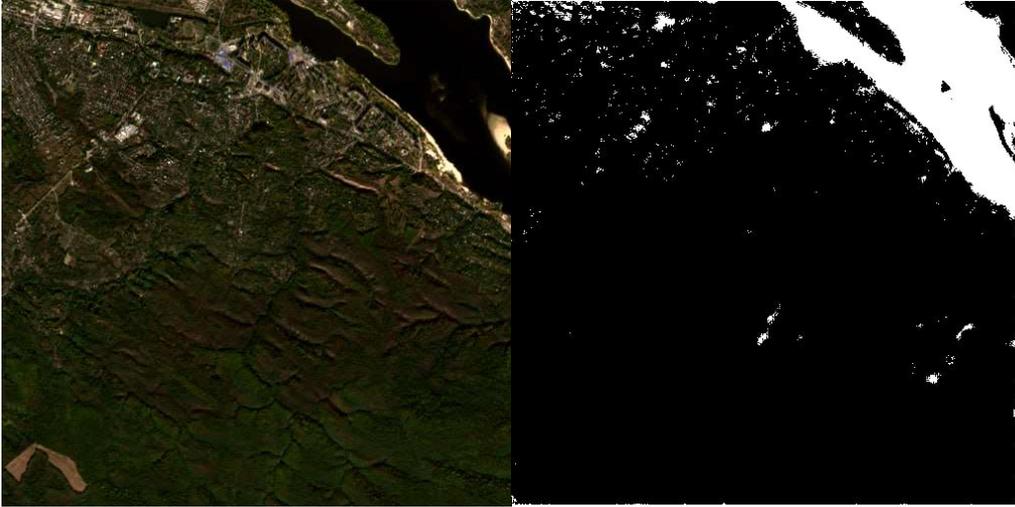


Figure 6: Example of UNetELU water segmentation result

PReLU

Parametric ReLU (PReLU) generalizes Leaky ReLU by learning the coefficient α during training rather than fixing it manually [26]. It is defined as:

$$f(x) = \begin{cases} x, & \text{for } > 0 \\ \alpha x, & \text{for } x \leq 0 \end{cases} \quad (4)$$

Here, α is initialized to a small positive value and optimized alongside other network parameters [8]. This adaptability allows the model to fine-tune the "leakiness" for each channel or neuron [14].

The main advantage of PReLU is its ability to dynamically adjust the negative slope to the data distribution, potentially improving accuracy [12]. However, it increases the number of parameters, necessitating stronger regularization. In water segmentation, PReLU may prove useful in cases where land-water boundaries are highly variable and require distinct sensitivity across channels [10].

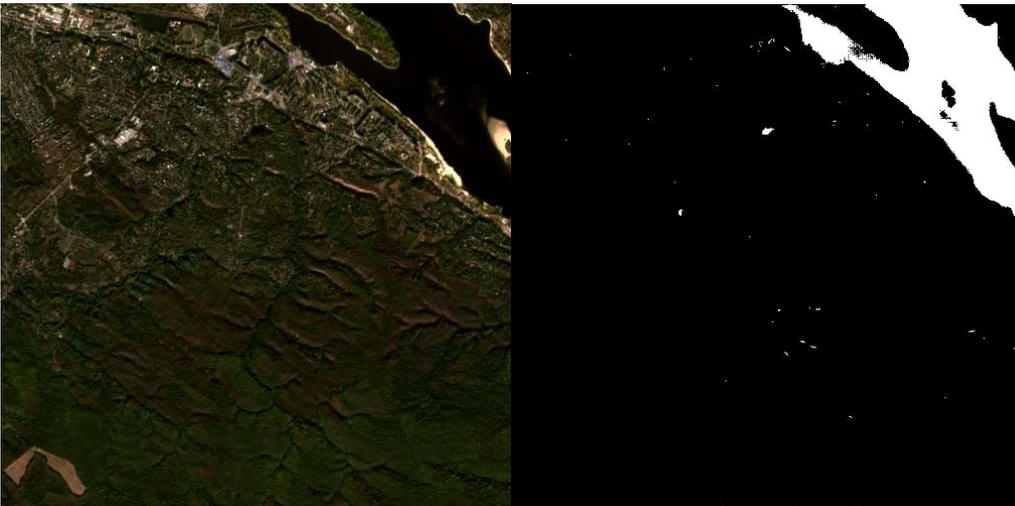


Figure 7: Example of UNetPReLU water segmentation result

Swish

Swish is considered a smoother alternative to ReLU that can enhance gradient flow in deep networks [20]. It is defined as:

$$f(x) = x\sigma(x), \quad \sigma(x) = \frac{1}{(1 + e^{-x})} \quad (5)$$

Unlike ReLU, Swish does not sharply zero-out negative values but smoothly attenuates them depending on the input magnitude [21]. This prevents neuron inactivation and may allow for better feature representation.

Swish avoids abrupt changes around $x = 0$, resulting in smoother gradients [4]. It has been shown to outperform ReLU in large-scale classification benchmarks such as ImageNet and COCO [22]. However, it is computationally more expensive due to the exponential calculations involved.

In this study, Swish is considered a promising alternative for scenarios where water-land boundaries are fuzzy or ill-defined. Its effectiveness, however, remains dependent on dataset size and training conditions [11].



Figure 8: Example of UNetSwish water segmentation result

RReLU

Randomized Leaky ReLU (RReLU) introduces stochasticity into the negative slope, sampling α from a specified range during training [14]. Formally:

$$f(x) = \begin{cases} x, & \text{for } x \geq 0 \\ \alpha_r x, & \text{for } x < 0 \end{cases}, \quad \alpha_r \in [a_{min}, a_{max}] \quad (6)$$

where α_r is a random value. This randomness can act as a regularizer, helping the model avoid overfitting or over-reliance on specific activation patterns [8]. However, it may also slow convergence if the variation range is too broad [16].

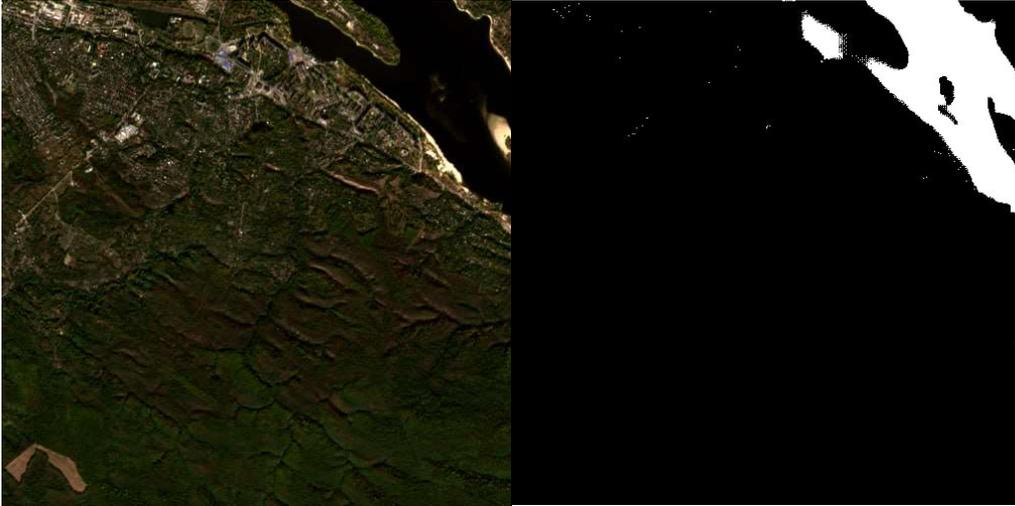


Figure 9: Example of UNetRRReLU water segmentation result

RReLU is potentially beneficial for heterogeneous datasets with varied conditions (e.g., seasonal differences, diverse lighting). This study explores whether its built-in variability leads to more generalized segmentation performance when compared to deterministic counterparts like Leaky ReLU and PReLU.

4. Results

The table below summarizes the comparative performance of all six activation functions based on four key evaluation metrics: F1 score, Precision, Recall, and Intersection over Union (IoU). Among the tested functions, **Leaky ReLU** achieved the highest F1 score (0.7386), along with the best precision (0.8395) and overall IoU (0.5856). **PReLU** ranked second in terms of F1 score (0.7253), showing a balanced performance with Precision of 0.7712 and Recall of 0.6845.

While **ELU** reached the highest Recall (0.7286), it suffered from low precision (0.5293), which led to the lowest overall F1 (0.6132) and IoU (0.4421) scores. **ReLU** and **Swish** produced similar mid-range results, with ReLU slightly outperforming Swish in Recall (0.7067 vs. 0.7028).

RReLU demonstrated relatively high precision (0.8261), exceeding that of PReLU, but its lower recall (0.6273) placed its F1 score (0.7131) and IoU (0.5541) between those of baseline ReLU and the top-performing Leaky ReLU. Thus, if the primary goal is to maximize F1 or IoU, Leaky ReLU is the most optimal. If recall is prioritized—for instance, to reduce false negatives in water detection—ELU may be considered, albeit at the cost of lower precision.

Table 1

Evaluation metrics of activation functions

Activation Function	F1 Score	Precision	Recall	IoU
ReLU	0.6881	0.6705	0.7067	0.5246
Leaky ReLU	0.7386	0.8395	0.6594	0.5856
ELU	0.6132	0.5293	0.7286	0.4421
PReLU	0.7253	0.7712	0.6845	0.5690
Swish	0.6792	0.6571	0.7028	0.5142
RReLU	0.7131	0.8261	0.6273	0.5541

5. Conclusions

The results of this comparative experiment confirm that the choice of activation function has a substantial impact on the performance of the U-Net model for binary segmentation of water bodies in Sentinel-2 satellite imagery. Leaky ReLU demonstrated the best overall results, achieving the highest values for F1 score and IoU while maintaining the strongest precision. PReLU followed closely, offering a balanced trade-off between precision and recall, though it still Leaky ReLU in all major metrics.

ELU stood out by achieving the highest recall, but this came at the expense of significant precision loss, resulting in the lowest F1 and IoU scores. The standard ReLU and Swish functions delivered average performance with no significant advantages over the more adaptive alternatives. RReLU offered high precision but somewhat reduced recall, placing its overall results between those of ReLU and Leaky ReLU.

In conclusion, for the segmentation of water surfaces from Sentinel-2 imagery, Leaky ReLU is the most effective activation function, offering the best balance between accuracy, completeness, and spatial consistency. In scenarios where maximizing recall is critical—such as minimizing omission of water pixels—ELU may be considered, albeit with a higher risk of false positives. To further enhance segmentation quality, future work should include expanding the training dataset with diverse geographic regions, optimizing activation-related hyperparameters, and exploring newer functions such as Mish or SELU, particularly in the context of multispectral Sentinel-2 data.

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-4o in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

- [1] Hnatushenko V., Honcharov O. Land Cover Mapping with Sentinel-2 Imagery Using Deep Learning Semantic Segmentation Models. CEUR Workshop Proceedings. Vol. 3909: Proc. of the XI International Scientific Conference "Information Technology and Implementation" (IT&I 2024), Kyiv, Ukraine, November 20–21, 2024. Kyiv, 2024. P. 1–18. <https://ceur-ws.org/Vol-3909/>
- [2] Soldatenko, D., Hnatushenko, Vik. Improving Satellite Imagery Recognition Performance with Initial Dataset Limitation by Augmenting Training Data. In: Babichev, S., Lytvynenko, V. (eds) Lecture Notes in Data Engineering, Computational Intelligence, and Decision-Making, Volume 2. ISDMCI 2024. Lecture Notes on Data Engineering and Communications Technologies, vol 244. Springer, Cham. https://doi.org/10.1007/978-3-031-88483-2_10
- [3] Hnatushenko, Vik., Hnatushenko, V., Soldatenko, D., and Heipke, C. Enhancing the quality of cnn-based burned area detection in satellite imagery through data augmentation, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLVIII-1/W2-2023, 1749–1755, <https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-1749-2023>
- [4] Goodfellow I., Bengio Y., Courville A. *Deep Learning*. – Cambridge, MA: MIT Press, 2016.
- [5] Isikdogan, Furkan & Bovik, Alan & Passalacqua, Paola. Surface Water Mapping by Deep Learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2017, pp. 1-10. 10.1109/JSTARS.2017.2735443.
- [6] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas. (). U-Net: Convolutional Networks for Biomedical Image Segmentation. *LNCS*, 9351, 2015, 234-241. 10.1007/978-3-319-24574-4_28.
- [7] Glorot X., Bordes A., Bengio Y. Deep Sparse Rectifier Neural Networks. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, PMLR 15:315-323, 2011. <https://proceedings.mlr.press/v15/glorot11a.html>

- [8] Maas, A.L., Hannun, A.Y. and Ng, A.Y. Rectifier Nonlinearities Improve Neural Network Acoustic Models. Proceedings of the 30th International Conference on Machine Learning, 2013, Vol. 28, 3. https://ai.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf
- [9] Hao Li, Johannes Zech, Christina Ludwig, Sascha Fendrich, Aurelie Shapiro, Michael Schultz, Alexander Zipf. Automatic mapping of national surface water with OpenStreetMap and Sentinel-2 MSI data using deep learning. International Journal of Applied Earth Observation and Geoinformation, Volume 104, 2021, <https://doi.org/10.1016/j.jag.2021.102571>.
- [10] Kampffmeyer, Michael & Salberg, Arnt-Børre & Jenssen, Robert. Semantic Segmentation of Small Objects and Modeling of Uncertainty in Urban Remote Sensing Images Using Deep Convolutional Neural Networks, 2016, 10.1109/CVPRW.2016.90.
- [11] Li, J., Cai, Y., Li, Q., Kou, M., & Zhang, T. A review of remote sensing image segmentation by deep learning methods. International Journal of Digital Earth, 17(1)6 2024. doi.org/10.1080/17538947.2024.2328827
- [12] Nair V., Hinton G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. Proceedings of the 27th International Conference on Machine Learning (ICML), 2010, USA, 807–814.
- [13] Chen Y., Jiang H., Li C., Jia X., Ghamisi P. Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks. IEEE Transactions on Geoscience and Remote Sensing, 2016, Vol. 54(10), pp. 6232–6251. doi: 10.1109/TGRS.2016.2584107
- [14] Xu B., Wang N., Chen T., Li M. Empirical Evaluation of Rectified Activations in Convolutional Network. 2015. arXiv:1505.00853
- [15] He K., Zhang X., Ren S., Sun J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. 2015. arXiv:1502.01852
- [16] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. & Lerer, A. Automatic Differentiation in PyTorch. NIPS 2017 Workshop on Autodiff, 2017. <https://openreview.net/forum?id=BJJsrmfCZ>
- [17] Clevert D.-A., Unterthiner T., Hochreiter S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs), 2015. arXiv:1511.07289
- [18] Mishkin D., Matas J. All you need is a good init. ICLR Workshop, 2016. doi.org/10.48550/arXiv.1511.06422
- [19] Goyal P., Dollár P., Girshick R., Noordhuis P., Wesolowski L., Kyrola A., Tulloch A. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. 2017. arXiv:1706.02677
- [20] Ramachandran P., Zoph B., Le Q.V. Searching for Activation Functions. 2018. arXiv:1710.05941
- [21] Elfving S., Uchibe E., Doya K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks. 107. 10.1016/j.neunet.2017.12.012.
- [22] Howard A., Sandler M., Chu G., et al. Searching for MobileNetV3. IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 1314-1324, doi: 10.1109/ICCV.2019.00140.
- [23] Kashtan V., Hnatushenko V. and Zhir S. Information Technology Analysis of Satellite Data for Land Irrigation Monitoring: Invited Paper. 2021 IEEE International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo), 2021, pp. 1-4, doi: 10.1109/UkrMiCo52950.2021.9716592.