Segmentation of Water Bodies Using Satellite Images and Deep Learning

Kawish Abbas
Department of Computer Science
National University of Computer and Emerging Sciences
Karachi, Pakistan
k225355@nu.edu.pk

ABSTRACT

This project presents a methodology for segmenting water bodies from Sentinel-2 satellite images using deep learning techniques. I employed a U-Net model and ViT to segment water bodies in Sindh, Pakistan, focusing on Halijee Lake, Hub Dam, Keenjhar Lake, and Manchar Lake. The model was trained and tested on images from 03/04/2023, 09/09/2019, and 17/05/2024. The results demonstrate the model's efficiency in accurately segmenting water bodies, which is crucial for environmental monitoring and management. The approach was tested on datasets consisting of water body images collected from Sentinel-2 satellite images, totaling over 5682 images. The U-Net model achieved training accuracy of 75.8% and validation accuracy of 73.5% whereas ViT model achieved accuracy of 74.1% and validation accuracy of 71.6%

I. INTRODUCTION

Water bodies extraction is critical in hydrology and water resources management. Accurate and up-to-date information on the location, size, and distribution of water bodies such as lakes, rivers, and wetlands can provide insights into the quantity and quality of available water resources. This information is important for the management of water resources, including water allocation, flood management, and water quality monitoring. Water body extraction plays an essential role in environmental studies, including ecology, biology, and geology. Accurate data on water bodies can provide insights into environmental conditions, habitat suitability, and species distribution. It is also important for climate change studies. Water bodies, particularly lakes, and reservoirs, play a critical role in the global carbon cycle and the exchange of greenhouse gases between the atmosphere and water bodies. Extracting accurate data on the size and distribution of water bodies is important for understanding the role of water bodies in the global carbon cycle, predicting the impacts of climate change on water resources, and designing effective adaptation strategies.[1]
Satellite imagery is a powerful tool for mapping water bodies.The availability of high-resolution satellite images and advanced image processing techniques can provide accurate and detailed information on water body dynamics and distribution.Traditional methods of water body segmentation involve manual interpretation and thresholding techniques, which can be time-consuming and less accurate. In this project, we explore the use of deep learning, specifically a U-Net, to automate the segmentation of water bodies from satellite imagery.

II. METHODOLOGY

A. Data Collection

Optical satellite systems have been used the most in water body extraction science. A collection of water bodies images captured by the Sentinel-2 Satellite. Each image comes with a black and white mask where white represents water and black represents something else but water. The masks were generated by calculating the NWDI (Normalized Water Difference Index) which is frequently used to detect and measure vegetation in satellite images, the dataset was downloaded for training [2].

B. Data Preprocessing

The preprocessing involves loading the images, converting them to the required format, and normalizing the pixel values.The dataset consists of satellite images and their corresponding masks. Each image and mask pair represents the presence of water bodies in the given satellite images.
The next step involves segregating the images and masks based on their labels.

```
for images, masks in data:
    X = images.numpy().astype("uint8")
    y = masks.numpy().astype("uint8")

print(X.shape, y.shape)

images = X[y == 0]
masks = X[y == 1]

print(images.shape, masks.shape)
```

Listing 1: Loading and Separating Images and Masks

Here, the images are separated based on their corresponding mask labels. The variable images contains all images where the mask label is 0, and masks contains all images where the mask label is 1. This segregation is crucial for ensuring that the images and masks are

correctly paired and processed.
To reduce the computational complexity, the RGB images and masks are converted to grayscale.For water body segmentation, the presence of water can often be determined by the intensity of the pixels rather than their color.

```python
X = np.zeros(shape=images.shape[:-1] + (1, ))
y = np.zeros(shape=masks.shape[:-1] + (1, ))

for i in range(X.shape[0]):
    X[i] = tf.image.rgb_to_grayscale(images[i])
    y[i] = tf.image.rgb_to_grayscale(masks[i])
```

Listing 2: Convert to Grayscale

Finally, the pixel values of the images and masks are normalized to a range of 0 to 1.Normalization is a common preprocessing step in deep learning. It ensures that the input data has a consistent scale, which can improve the convergence of the model during training. In this case, the pixel values, initially ranging from 0 to 255, are divided by 255 to bring them into the [0, 1] range.

```python
X = X / 255.0
y = y / 255.0
```

Listing 3: Normalization

C. Threshold Example

Thresholding is a fundamental technique in image processing used to create binary images from grayscale images. By applying a threshold value, each pixel in the image is converted to either black or white, depending on whether its intensity is below or above the threshold. In the context of satellite images, thresholding can be used to distinguish between different features such as water bodies, vegetation, urban areas, and so on.

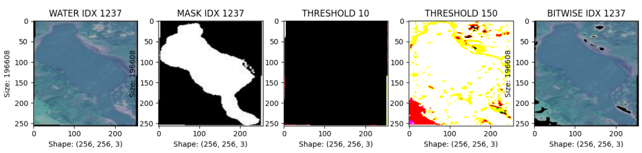


Fig. 1: Threshold

D. U-Net for Water Bodies Extraction

The U-Net architecture is designed for precise image segmentation, particularly for biomedical images, but its effectiveness extends to various other domains, including satellite imagery for tasks like water body segmentation. It is characterized by a symmetric, U-shaped structure composed of an encoder (contracting path) and a decoder (expansive path), connected by skip connections.

1) Encoder:
The encoder path captures the context and features from the input image through a series of convolutional and pooling layers. It consists of the following steps:
   a) Convolutional Layers:
   Each encoder block contains two convolutional layers with a small 3x3 kernel size, followed by a nonlinear activation function (e.g., ReLU or ELU).
   b) Pooling Layer:
   A 2x2 max pooling layer follows the convolutional layers, which reduces the spatial dimensions of the feature maps by a factor of 2. This pooling operation helps to capture hierarchical features and reduces computational complexity.
   c) Feature Map:
   The number of feature maps (or filters) doubles after each pooling layer, allowing the network to learn increasingly complex features at each level
2) Bottleneck:
At the deepest part of the U-Net, the bottleneck layer connects the encoder and the decoder. This layer has the following characteristics:
   a) Convolutional Layers:
   Similar to the encoder blocks, the bottleneck consists of two convolutional layers with a small 3x3 kernel size, followed by an activation function.
   b) High-Level Features:
   The bottleneck captures the most abstract and high-level features of the input image, serving as a bridge between the contracting and expansive paths.
3) Decoder:
The decoder path reconstructs the spatial dimensions of the input image and produces a dense segmentation map. It consists of the following steps:
   a) Upsampling Layers:
   Each decoder block begins with an upsampling operation (often implemented as a transposed convolutional layer), which increases the spatial dimensions of the feature maps by a factor of 2.
   b) Concatenation with Skip Connections:
   Feature maps from the corresponding encoder block are concatenated with the upsampled feature maps. These skip connections allow the decoder to access high-resolution features from the encoder, which helps in preserving fine details in the segmentation output.
   c) Convolutional Layers:
   Similar to the encoder, each decoder block

contains two convolutional layers with a small 3x3 kernel size, followed by an activation function.

d) Feature Maps:
The number of feature maps is halved after each upsampling layer, mirroring the progression in the encoder.

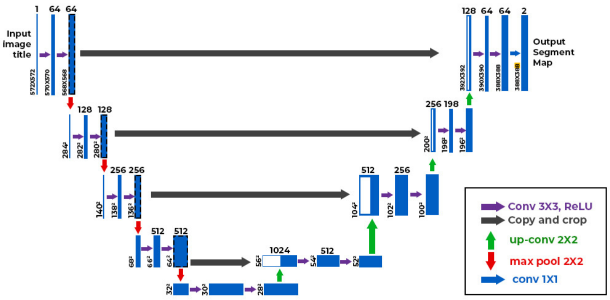


Fig. 2: U-Net architecture

E. Training, Validation, and Testing:

It's crucial to properly split your data into training, validation, and testing sets. This ensures that your model learns from a diverse range of examples, generalizes well to unseen data, and can accurately evaluate its performance. The training and evaluation datasets are used to train the model, while the testing dataset is used to assess the model on unseen data. Unseen data are applied for simulating real-world prediction, as the model has not seen this data before.The dataset is partitioned into two subsets, with 80% allocated for training, 20% for validation,for testing, I have utlized Sentinel 2 Images of Sindh comprising RGB images and NDWI images [3]. During training, the network is set to run 50 epochs.

$$NDWI = \frac{G - NIR}{G + NIR} \quad (1)$$

The Normalized Difference Water Index (NDWI) is derived from the Near-Infrared (NIR) and Green (G) channels. This formula highlights the amount of water in water bodies. An alternate method of calculation uses the NIR and Short Wave Infrared (SWIR) channels [(NIR-SWIR)/(NIR+SWIR)]

F. The loss and accuracy for the training and validation datasets

A loss function is used to help a machine-learning algorithm to optimize. The loss is based on training and validation, and how well the model is performing in these two test sets. It is the sum of all mistakes made in all training and test sets. A loss estimate is a method that describes how badly or well the model behaves after each cycle of optimization.

III. Results and Discussion

Satellite image classification was executed to characterize the classes of (water and no-water), based on U-Net.

A. Evaluation

The performance of the U-Net model and ViT for water body segmentation from satellite images was evaluated using accuracy as the primary metric. Accuracy is the ratio of correctly predicted pixels (both water and non-water) to the total number of pixels. It gives a straightforward measure of the overall performance of the model:

$$Accuracy = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Pixels}} \quad (2)$$

The training accuracy of 75.81% suggests that the model has learned to identify water bodies with a reasonably high degree of precision on the training dataset. The validation accuracy of 73.50% shows that the model maintains a similar level of performance on unseen data, which indicates that the U-Net model has generalization capabilities and is not overly fitted to the training set as compare to ViT model.

TABLE I: Performance Metrics for U-Net and ViT Models

Model	Training Accuracy	Validation Accuracy	Training Loss	Validation
U-Net	0.7580	0.7350	0.2812	0.3224
ViT	0.7411	0.7169	0.3305	0.3738

Overall, the U-Net model exhibits promising results in the task of water body segmentation. With further fine-tuning and optimization, the model's performance can be enhanced, making it a valuable tool for environmental monitoring and management based on satellite imagery

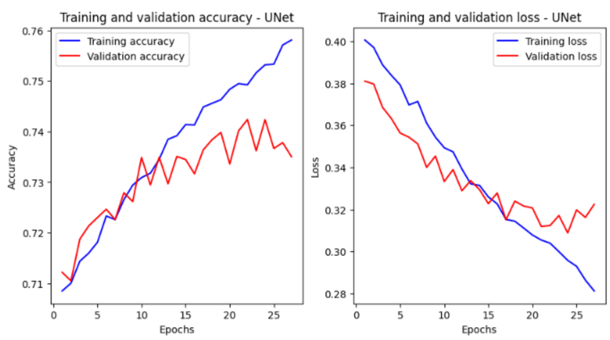


Fig. 3: U-Net Model Evaluation

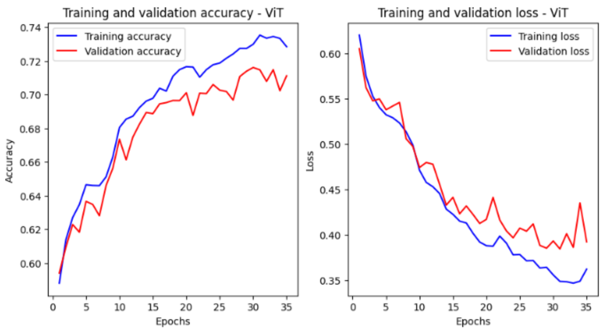


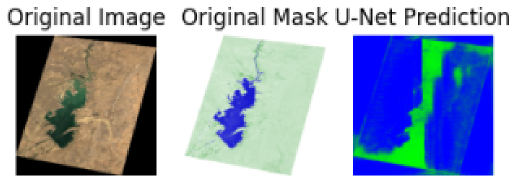
Fig. 4: ViT Model Evaluation

B. Prediction on Test Data

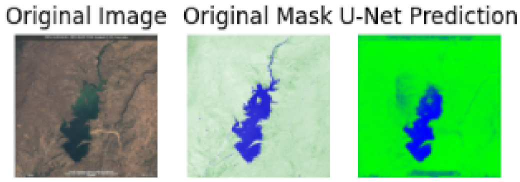
To evaluate the model's performance on new images, I downloaded satellite images of Keenjhar Lake, Hub Dam, Haleji Lake, and Manchar Lake. These images include both true color and NDWI (Normalized Difference Water Index) representations to observe the differences. The images were acquired on different dates to capture temporal changes: May 17, 2024, September 9, 2019, and April 3, 2023

TABLE II: Acquisition Dates for Satellite Images

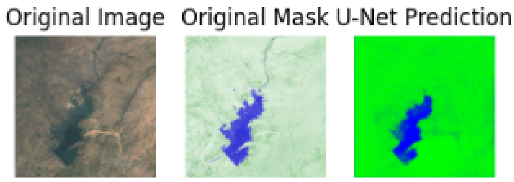
Location	True Color Image Date	NDWI Image Date
Keenjhar Lake	17/05/2024	17/05/2024
Hub Dam	09/09/2019	09/09/2019
Haleji Lake	03/04/2023	03/04/2023
Manchar Lake	17/05/2024	17/05/2024



(a) Hub Dam 17 May 2024

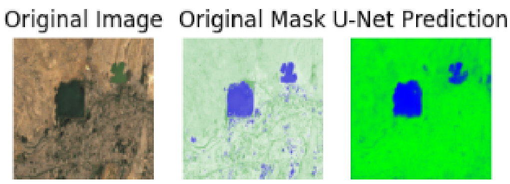


(b) Hub Dam 03 April 2023

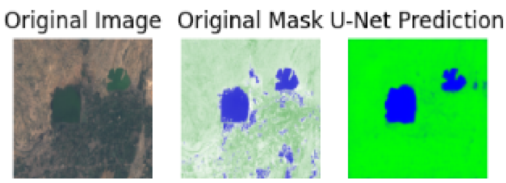


(c) Hub Dam 09 Sep 2019

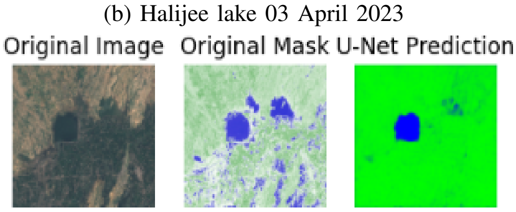
Fig. 6: U-Net Prediction Hub Dam



(a) Halijee lake 17 May 2024

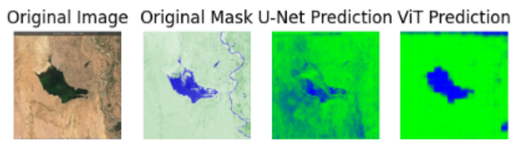


(b) Halijee lake 03 April 2023

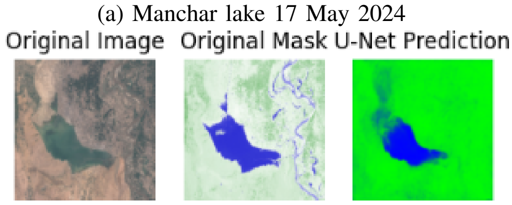


(c) Halijee lake 09 Sep 2019

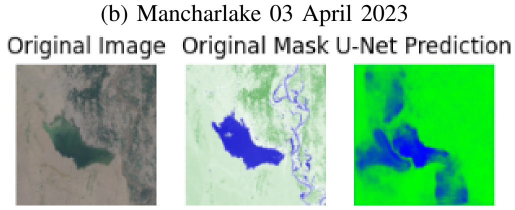
Fig. 5: U-Net Prediction Halijee lake



(a) Manchar lake 17 May 2024

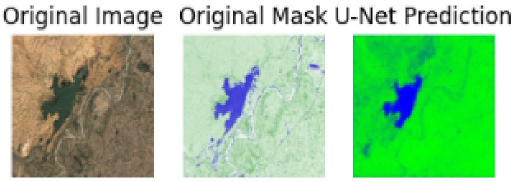


(b) Mancharlake 03 April 2023

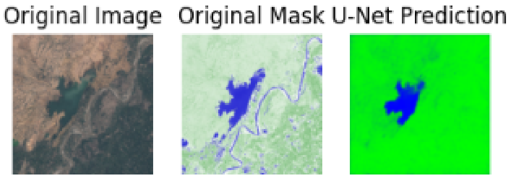


(c) Manchar lake 09 Sep 2019

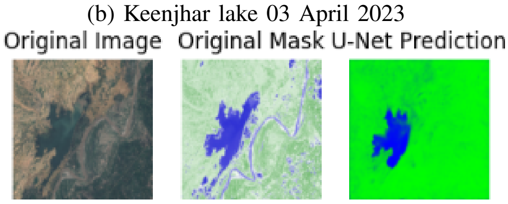
Fig. 7: U-Net Prediction Manchar lake



(a) keenjhar lake 17 May 2024



(b) Keenjhar lake 03 April 2023



(c) Keenjhar lake 09 Sep 2019

Fig. 8: U-Net Prediction Keenjhar lake

Overall, the U-Net model exhibits promising results in the task of water body segmentation. With further fine-tuning and optimization, the model's performance can be enhanced, making it a valuable tool for environmental monitoring and management based on satellite imagery.

IV. Conclusion

In this study, we presented a methodology for segmenting water bodies from Sentinel-2 satellite images using deep learning techniques. We employed a U-Net model and ViT for this task, focusing on water bodies in Sindh, Pakistan. The models were trained and evaluated on various images, demonstrating their efficiency in accurately segmenting water bodies.

The U-Net model achieved a training accuracy of 75.8% and a validation accuracy of 73.5%, while the ViT model achieved an accuracy of 74.1% and a validation accuracy of 71.6%. These results indicate that both models perform well in identifying water bodies from satellite imagery.

Furthermore, the evaluation of the models on test data from different dates and locations, such as Keenjhar Lake, Hub Dam, Halijee Lake, and Manchar Lake, showed promising results. The models successfully identified water bodies in these images, demonstrating their robustness and generalization capabilities.

A. Future Work

In future work, I plan to explore techniques for further improving the performance of the models, such as data augmentation, model ensembling, and transfer learning. Additionally, we aim to apply the models to larger geographic regions and evaluate their performance in different environmental conditions and seasons.

References

[1] M. E. Enan, "Deep learning for studying urban water bodies spatio-temporal transformation," 2021.
[2] https://www.kaggle.com/datasets/franciscoescobar/satellite-images-of-water bodies, "Satellite images of water bodies."
[3] https://apps.sentinel hub.com/, "Sentinel hub."