RECONNAISSANCE IN NATURAL ENVIRONMENTS II
Image Data Generation

Pauly Alexandre (312551810)
Clémenceau Maxime (312551812)

University of NYCU, Hsinchu

ABSTRACT

This report describes the work conducted as part of the project carried out at NYCU in Hsinchu, Taiwan, during the spring semester of 2024, initiated by the AI CUP.

The objective of this competition is to use generative AI techniques to create precise delineation lines of waterways and roads from images captured by drones.

In this report, we will explore the challenges encountered during the development of such a solution.

I. INTRODUCTION

Technological advancements in the field of artificial intelligence have opened up new perspectives in various sectors, including the recognition and monitoring of natural environments. However, to fully harness these advancements, it remains essential to be able to guide autonomous systems accurately and efficiently.

It is in this context that competitions such as the AI CUP have emerged as crucial platforms to stimulate innovation and promote the advancement of AI techniques. Our participation in this competition aimed to develop a generative artificial intelligence model for information and recognition in natural environments. Specifically, we focused on creating precise delineation lines of waterways and roads from images captured by drones.

By combining artificial intelligence techniques with image processing, our team tackled this exciting challenge, seeking to push the boundaries of visual perception in autonomous systems. The first part of the report will focus on contextualizing the project, providing an analysis of the state of the art in the field and presenting the objectives and challenges that guided our work.

We will also explore the technical and conceptual challenges encountered throughout the development process, as well as the solutions we proposed to overcome them. Finally, we will highlight the lessons learned during this enriching experience, aiming to draw insights for future projects and improvements.

II. RELATED WORKS & BACKGROUND

In recent years, the field of image processing and artificial intelligence (AI) has witnessed significant advancements, particularly in tasks related to semantic segmentation and boundary detection. Numerous studies have explored various approaches to tackle similar challenges to ours, aiming to extract precise features from images captured by drones.

One prominent area of research is the application of convolutional neural networks (CNNs) for semantic segmentation tasks. Models such as U-Net, SegNet, and DeepLab have demonstrated remarkable performance in accurately delineating object boundaries in images. These networks leverage the hierarchical structure of CNNs to capture both local and global contextual information, enabling them to generate detailed segmentation maps.

Additionally, the use of generative adversarial networks (GANs) has shown promise in image synthesis tasks. The paper “Image to Image Translation with Conditional Adversarial Networks” [1] demonstrates how conditional GANs can be employed for tasks such as generating edge maps from images. By training a GAN on a dataset of drone images containing both road and river segments, the generator network can learn to produce realistic boundary maps, while the discriminator network ensures the generated boundaries align with the features present in the original images.

Furthermore, incorporating attention mechanisms into neural network architectures has proven beneficial for tasks requiring precise localization, such as boundary detection. The paper “Attention U-Net: Learning Where to Look for the Pancreas” [2] introduces an attention mechanism in the U-Net architecture, enhancing its ability to focus on relevant regions of the input image.

Semantic segmentation, a fundamental task in computer vision, involves partitioning an image into multiple semantically meaningful regions. Accurate boundary detection is crucial for various applications, including urban planning, environmental monitoring, and infrastructure development.

Traditional methods for boundary detection often rely on handcrafted features and heuristic algorithms, which may struggle to generalize across diverse environmental conditions and image characteristics. In contrast, deep learning approaches have emerged as a powerful tool for boundary detection, capable of learning complex representations directly from raw image data.

Convolutional neural networks (CNNs) have revolutionized the field of computer vision, achieving state-of-the-art performance in various tasks, including semantic segmentation and object detection. Generative adversarial networks (GANs) represent another breakthrough in deep learning, particularly in the domain of image synthesis.

Attention mechanisms have garnered increasing attention in the deep learning community for their ability to improve model interpretability and performance.

By leveraging these advanced techniques and building upon prior research in image processing and AI, we aim to develop a robust and efficient model for generating precise boundary maps of rivers and roads.

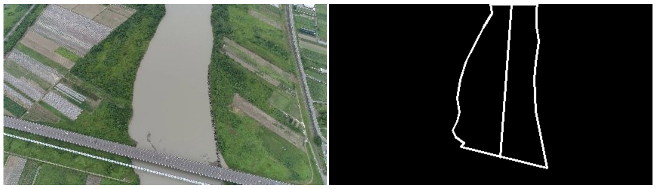
III. DATASETS

The dataset for this project is designed to facilitate the training and evaluation of generative AI models for UAV imagery of roads and rivers. It is divided into training, public testing, and private testing datasets, each with specific formats and purposes.

The training dataset contains two folders: *img/* and *label_img/*. The *img/* folder holds images in *.jpg* format, while the *label_img/* folder contains corresponding labels in *.png* format.

Files are named using the format TRA_XX_XXXXXXX, where XX indicates the type of data (RI for rivers and RO for roads) and XXXXXXX is a sequential serial number starting from 0. Files with the same name but different extensions (e.g., TRA_RI_1000000.jpg and TRA_RI_1000000.png) form a single dataset.

The *.jpg* files contain raw UAV images, while the *.png* files are black and white, with white lines indicating the boundaries and centerlines of roads or rivers. All images have a resolution of 428x240 pixels. The training dataset includes 4320 datasets, equally divided between rivers and roads.

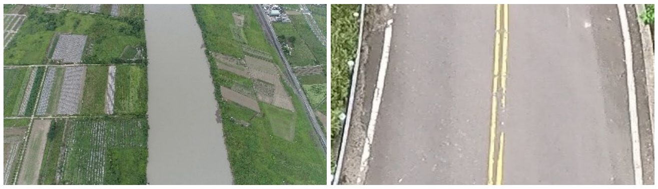


TRA_RI_1000000.jpg *TRA_RI_1000000.png*

Fig. 1. Training Dataset Illustration

The public testing dataset contains 720 black and white images in *.png* format. Files follow the format PUB_XX_XXXXXXX, where XX indicates the type of data to generate (RI for rivers and RO for roads) and XXXXXXX is a sequential serial number starting from 0.

Similar to the training dataset, the images are 428x240 pixels, with white lines representing boundaries and centerlines. This dataset includes 720 data entries, split equally between rivers and roads.

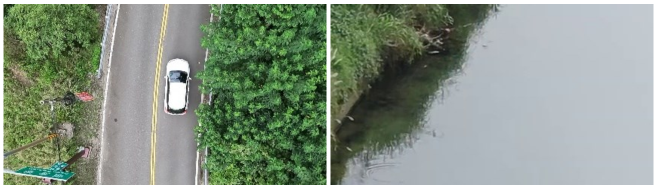


PUB_RI_2000004.jpg *PUB_RO_2000005.jpg*

Fig. 2. Testing Dataset Illustration

The private testing dataset, like the public testing dataset, contains 720 black and white images in *.png* format. Files are named using the format PRI_XX_XXXXXXX, where XX indicates the type of data to generate (RI for rivers and RO for roads) and XXXXXXX is a sequential serial number starting from 0.

The images are consistent in size and format with the training and public testing datasets, being 428x240 pixels with white lines indicating boundaries and centerlines. This dataset also includes 720 data entries, equally divided between rivers and roads.



PUB_RO_2000162.jpg *PUB_RI_2000051.jpg*

Fig. 3. Private Dataset Illustration

IV. LEARNING METHODS

In our project, we explored three distinct convolutional neural network (CNN) architectures to identify and delineate the boundaries of roads and rivers in UAV imagery: U-Net, a deeper variant of U-Net, and V-Net.

Each of these architectures offers unique characteristics that make them suitable for the challenging task of segmenting high-resolution UAV images.

A. U-Net

The U-Net architecture, originally developed for biomedical image segmentation, features a symmetric encoder-decoder structure with skip connections that transfer feature maps from the encoder to the decoder, preserving fine-grained details.

Our implementation of U-Net included four convolutional layers in both the encoder and decoder. The encoder captured context through a series of convolutional and pooling layers, which progressively reduced the spatial dimensions while increasing the depth of feature maps. The decoder, on the other hand, reconstructed the spatial resolution by up sampling the feature maps and applying further convolutions. The skip connections allowed the model to retain high-resolution information from the encoder, crucial for precise boundary detection.

This balanced architecture of U-Net is efficient for learning from limited data, which is often the case in UAV imagery, making it a suitable choice for mapping road and river boundaries.

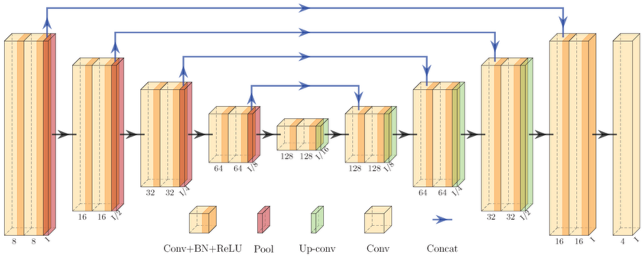


Fig. 4. U-Net Implementation

B. U-Net Deeper

To further enhance the segmentation performance, we experimented with a deeper variant of U-Net.

This deeper architecture incorporated additional convolutional layers and down sampling / up sampling operations, increasing the network's depth to six layers in both the encoder and decoder.

By adding more convolutional filters at each layer, the deeper U-Net aimed to capture more complex features and finer details. This deeper structure allowed the network to learn more intricate patterns and representations, improving its ability to generalize and accurately segment images with varying levels of complexity.

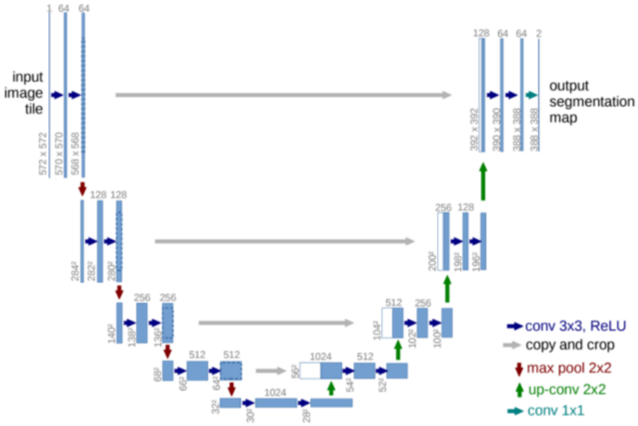


Fig. 5. Deeper U-Net Implementation

C. V-Net

V-Net, initially designed for volumetric medical image segmentation, was adapted for 2D UAV imagery in our project. V-Net features a volumetric fully convolutional network with residual connections that facilitate the training of very deep networks by allowing the gradient to flow more smoothly through the layers.

Our adaptation of V-Net included five residual blocks in both the encoder and decoder. The use of residual connections in V-Net helped stabilize the training process and prevented the degradation problem, where deeper networks fail to train effectively.

This architecture's ability to maintain high accuracy over many layers made it effective for capturing the complex structures and boundaries present in UAV images of roads and rivers. V-Net's design allowed for a robust extraction of features across various scales, crucial for accurately delineating intricate boundaries in our UAV datasets.

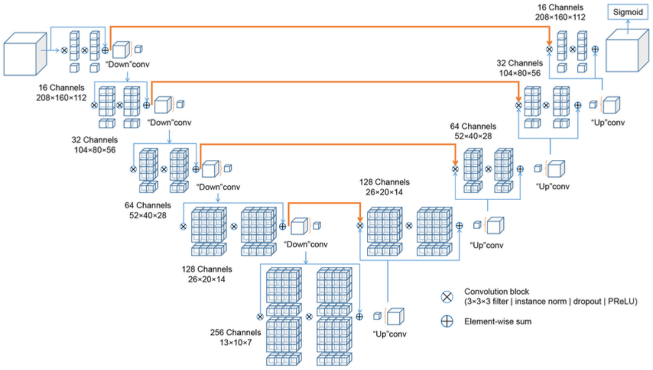


Fig. 6. V-Net Implementation

By leveraging these three architectures: U-Net for its balanced and efficient learning, the deeper U-Net for its enhanced capacity for feature learning, and V-Net for its stable training of deeper networks, we aimed to achieve high fidelity in segmenting road and river boundaries in UAV imagery.

Each architecture brought its strengths to the table, allowing us to explore a range of approaches to improve the segmentation accuracy and robustness in our challenging dataset.

V. POST TREATEMENT METHODS

In the post-processing stage of our river/road boundary detection project, we employ several techniques to refine and enhance the accuracy of our model's predictions.

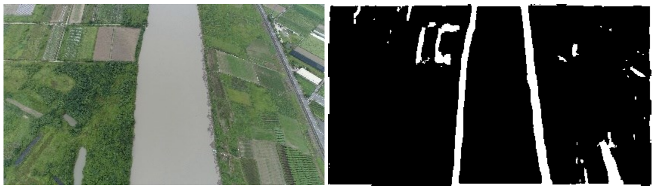


Fig. 7. Result after using U-Net Model

One fundamental method is the application of opening and closing operations.

Opening involves the sequential application of erosion followed by dilation, which effectively removes small noise or irregularities from the image. This step is crucial because aerial or satellite imagery often contains various artifacts or small features that can interfere with accurate boundary detection.

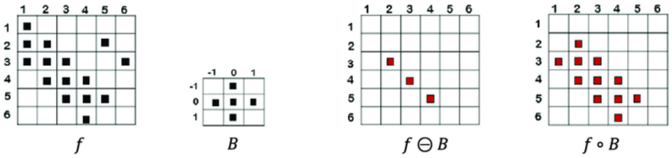


Fig. 7. Opening Method Illustration with 3x3 kernel

Closing, on the other hand, reverses this process by first dilating and then eroding the image. This operation helps to bridge small gaps or holes in the boundaries, ensuring smoother and more continuous delineations.

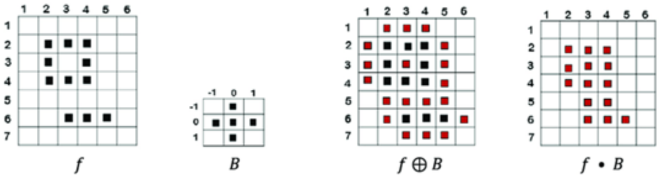


Fig. 8. Closing Method Illustration with 3x3 kernel

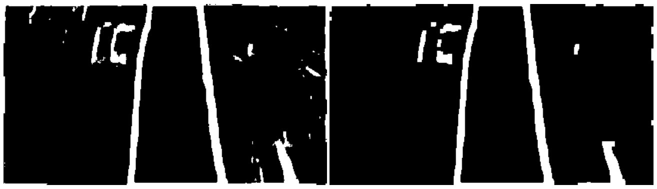


Fig. 8. Result after Opening / Closing with 5x5 kernel

Additionally, we implement a function to extract the largest connected component from the image. Roads and rivers typically form large, contiguous features in aerial imagery, with smaller components often representing noise or insignificant details.

By retaining the largest component, we focus our analysis on the primary features of interest, effectively filtering out minor artifacts and clutter that could otherwise obscure the boundaries.

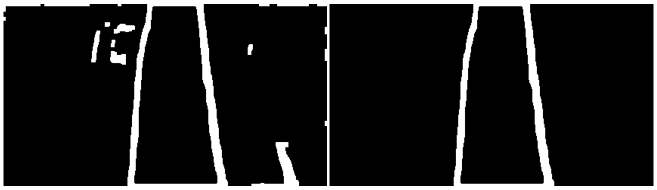


Fig. 9. Result after retaining the Largest Component

To further refine the detected boundaries and reduce their thickness, we apply a Laplacian filter. This filter calculates the second derivative of the image intensity, enhancing abrupt intensity changes associated with edges and boundaries.

By subtracting the Laplacian image from the original, we effectively sharpen the edges of the detected features, resulting in clearer and more precise delineations of roads and rivers.

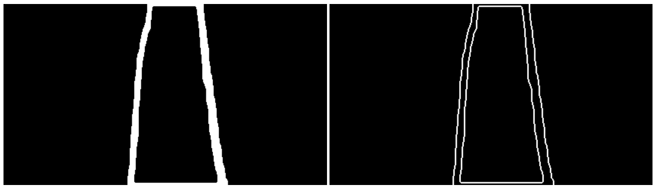


Fig. 10. Laplacian Filter Result

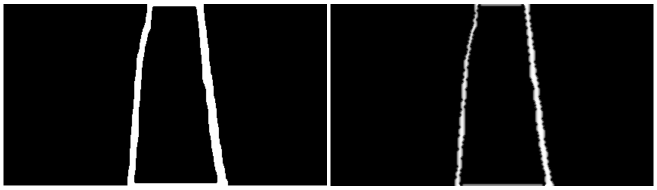


Fig. 11. Multiple Iteration of Laplacian Filter Result

Moreover, we employ the Canny edge detector, a multi-stage algorithm renowned for its effectiveness in edge detection tasks.

The Canny filter begins by smoothing the image to reduce noise, then identifies areas with significant intensity changes using gradient calculation. Subsequent steps involve non-maximum suppression and hysteresis thresholding to detect and link edges accurately.

This method is particularly valuable for capturing the intricate details of road and river boundaries with high precision.

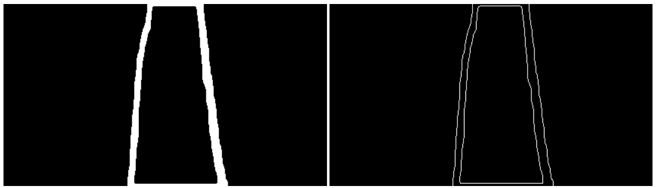


Fig. 12. Canny Edge Detector Result

While the HoughLinesP line detection tool is a valuable resource for detecting straight lines in images, its utility may be limited in cases where roads or rivers exhibit non-linear or curved boundaries.

Therefore, while we acknowledge its effectiveness in certain scenarios, we primarily rely on other techniques to ensure comprehensive boundary detection in our aerial imagery.

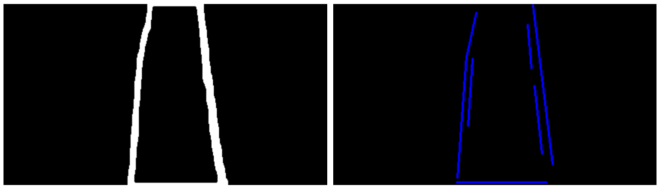


Fig. 13. HoughLinesP Line Detection Result

The final step of our post-treatment is to add a line to the middle of the predicted river or road in each image. This enhancement serves to emphasize the central axis of these boundaries, providing a clear and concise representation of the detected features.

To achieve this, we first identify the centerline of the predicted boundary using morphological operations and edge detection techniques. This process involves thinning the boundary to a single-pixel width, ensuring an accurate representation of the central path.

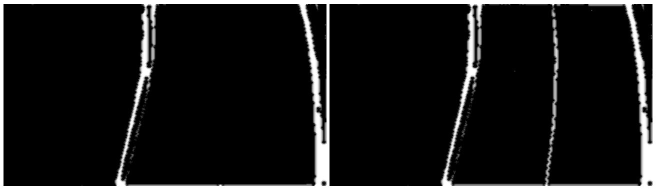


Fig. 14. Middle Line Creation Result

By integrating these post-processing methods into our workflow, we enhance the clarity, accuracy, and reliability of our river/road boundary detection system, ultimately yielding more actionable insights for various applications, from urban planning to environmental monitoring.

VI. METRICS

In assessing the performance of our river/road boundary detection models, we employed the F-measure metric, a widely used evaluation method in image analysis tasks.

This metric combines precision and recall, providing a balanced assessment of model performance. To calculate this metric, we need to calculate the TP, FP and FN values. True Positives (TP) represent white pixels correctly identified in both the Ground Truth and the generated image, while False Positives (FP) denote white pixels in the generated image but black pixels in the Ground Truth. False Negatives (FN) are white pixels in the Ground Truth but black pixels in the generated image. With these three components which indicate evaluate the performance of our algorithm:

Recall, also known as sensitivity, measures the ability of a model to correctly identify all relevant instances of a class. In our context, it signifies the proportion of true positives (correctly identified white pixels) out of all actual positives (total white pixels in the Ground Truth). A high recall indicates that the model is adept at capturing most of the relevant information, minimizing the number of false negatives.

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

Fig. 15. Recall Formula

Precision quantifies the accuracy of positive predictions made by the model. It represents the proportion of true positives (correctly identified white pixels) out of all predicted positives (total white pixels in the generated image). Precision focuses on minimizing false positives, ensuring that the model's predictions are reliable and trustworthy.

$$Precision = \frac{TP}{(TP + FP)}$$

Fig. 16. Precision Formula

The F-score, also known as the F1-score, is a harmonic mean of precision and recall. It provides a balanced assessment of a model's performance by considering both false positives and false negatives. The F-score is particularly useful when there is an imbalance between precision and recall, as it combines these metrics into a single value. In our case, we adjusted the F-score calculation using β²=0.3 to place more emphasis on precision, which is often critical in boundary detection tasks.

$$F - score = \frac{\left(1+\beta^2\right) Recall * Precision}{\beta^2 Precision + Recall}$$

Fig. 17. F-Score Formula

To achieve a better-performing model, we aim to maximize both recall and precision while optimizing the F-score. However, there's often a trade-off between recall and precision improving one may negatively impact the other.

Therefore, finding the optimal balance between these metrics is crucial. In general, a higher recall indicates that the model captures more relevant information, while a higher precision implies fewer false positives. Ideally, we strive for high values in both metrics to ensure that our model accurately identifies river and road boundaries while minimizing errors. Similarly, a higher F-score indicates a more balanced performance between precision and recall, reflecting an overall better-performing model.

Therefore, in our evaluation process, we aim to achieve high values for recall, precision, and F-score to validate the effectiveness of our boundary detection models.

VII. RESULTS AND ANALYSIS

In the initial implementation of our river and road boundary detection model, we opted for the U-Net architecture due to its proven efficacy in semantic segmentation tasks.

However, during the training phase, we encountered a significant challenge: the model consistently predicted only black pixels, but still achieving a good accuracy. Upon investigation, we identified a class imbalance issue wherein the dataset contained a disproportionately large number of black pixels compared to white pixels. As a result, the model favored predicting black pixels to optimize accuracy, neglecting the detection of white pixels representing the boundaries of rivers and roads.

To address this issue and ensure more balanced predictions, we conducted an extensive analysis of the dataset, examining the distribution of white and black pixels across a subset of 1000 images. Leveraging this data, we devised a strategy to calculate the optimal weights to assign to our models, thereby mitigating the impact of class imbalance. Subsequently, we developed a custom loss function tailored to our specific dataset characteristics, aiming to incentivize the model to accurately detect both white and black pixels, thus improving the overall performance and reliability of our boundary detection system.

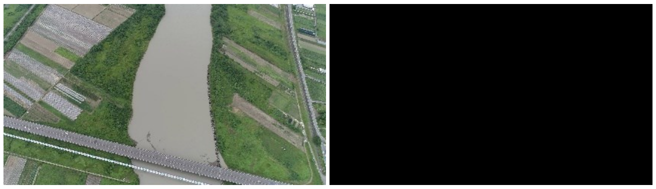
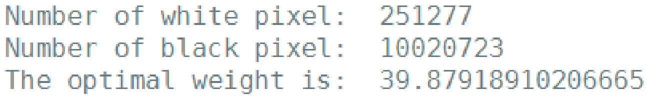


Fig. 18. Prediction without the Optimal Weight



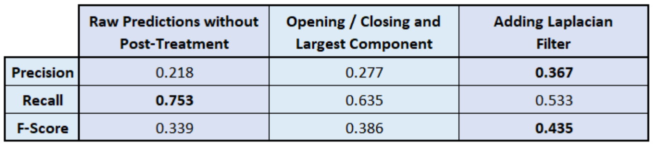
```
Number of white pixel:  251277
Number of black pixel:  10020723
The optimal weight is:  39.87918910206665
```

Fig. 19. Optimal Weight Calculated

In our first set of experiments, we aimed to achieve the best results using the U-Net method by applying various post-treatment techniques.

Initially, we evaluated the raw predictions from the U-Net model without any post-treatment to establish a baseline. Following this, we introduced a combination of opening and closing operations to remove noise, followed by extracting the largest connected component to focus on the main features of interest, such as rivers and roads. Lastly, we applied the Laplacian filter to these refined images to reduce the thickness of the detected boundaries, enhancing their precision.

By systematically comparing the outcomes of these different post-processing approaches, we sought to determine the most effective method for improving the accuracy and clarity of our river and road boundary detections.



	Raw Predictions without Post-Treatment	Opening / Closing and Largest Component	Adding Laplacian Filter
Precision	0.218	0.277	**0.367**
Recall	**0.753**	0.635	0.533
F-Score	0.339	0.386	**0.435**

Fig. 20. Results without different Post Treatment on U-Net

Through these systematic comparisons, we found that the best results were obtained when applying the full suite of post treatment techniques, including opening and closing operations, largest component extraction, and the Laplacian filter. This comprehensive approach significantly improved the accuracy and clarity of our river and road boundary detections.

After refining our post-processing techniques, we proceeded to compare the performance of the two different models used: U-Net and V-Net.

Our objective was to determine which architecture would yield superior results for river and road boundary detection. Both models are designed for semantic segmentation tasks, but they differ in their structural approaches.

U-Net features a symmetric encoder-decoder architecture with skip connections, preserving high-resolution information throughout the network.

In contrast, V-Net employs a volumetric approach with residual connections, potentially offering better performance in capturing complex structures.

By training and evaluating both models under identical conditions and applying the same post-processing techniques, we aimed to identify the model that best balances accuracy and efficiency for our specific application.

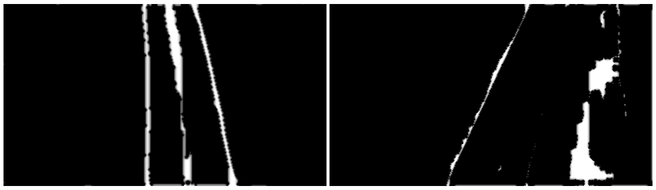


Fig. 21. River and Road Prediction with V-Net, 10 epochs and a batch size equal to 5

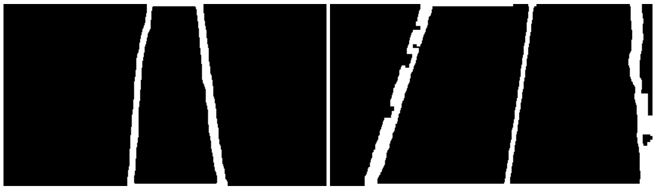
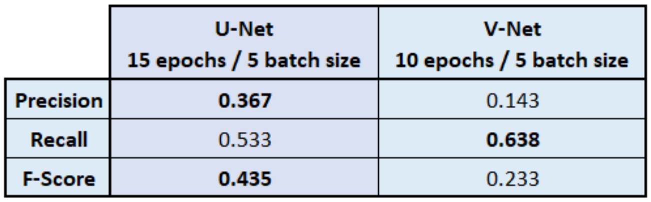


Fig. 22. River and Road Prediction with U-Net, 15 epochs and a batch size equal to 5



	U-Net 15 epochs / 5 batch size	V-Net 10 epochs / 5 batch size
Precision	**0.367**	0.143
Recall	0.533	**0.638**
F-Score	**0.435**	0.233

Fig. 23. Results for both Model after Post Treatment

After analysis and evaluation, we determined that the U-Net model consistently outperformed the V-Net model in our river and road boundary detection tasks. Despite V-Net's advanced volumetric approach, U-Net's symmetric encoder-decoder architecture with skip connections proved more effective in preserving high-resolution details and accurately delineating boundaries. Therefore, we concluded that U-Net remains the superior choice for our specific application.

With U-Net model and our post treatment method, our next step was to enhance the clarity of the detected boundaries by predicting the middle line of the rivers and roads.

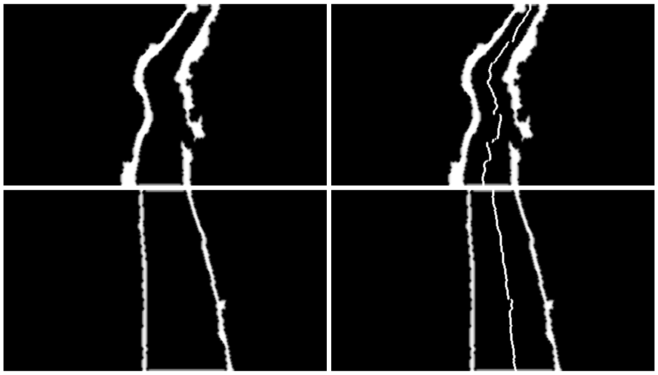
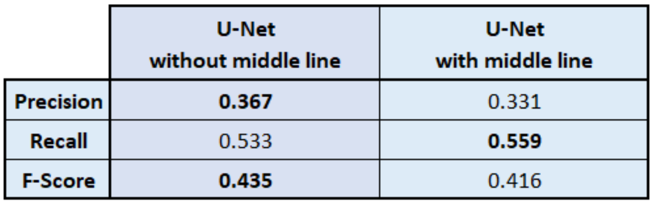


Fig. 24. Middle Line Prediction on U-Net Post Treated Images

We can observe that the middle line prediction seems to work well, clearly highlighting the central path of the detected rivers and roads. This enhancement provides a more precise and useful representation for mapping and navigation purposes.

However, to ensure the effectiveness and accuracy of this approach, we now need to verify the results using our established metrics.



	U-Net without middle line	U-Net with middle line
Precision	**0.367**	0.331
Recall	0.533	**0.559**
F-Score	**0.435**	0.416

Fig. 25. Results for U-Net Model with / without the Middle Line

It seems that the images with the predicted middle line are exhibiting a lower F-score compared to those without it.

One possible reason for this discrepancy could be that the process of thinning the boundaries to a single-pixel width might inadvertently remove some of the finer details, leading to a loss of crucial information that affects the F-score.

Additionally, the middle line prediction might introduce artifacts or slight inaccuracies that the F-score metric, which is sensitive to both precision and recall, penalizes more heavily.

Another potential factor could be that the middle line, while visually helpful, might not align perfectly with the ground truth boundaries used for metric evaluation, resulting in a lower score.

These factors suggest that while the middle line improves visual clarity, it may need further refinement to maintain high quantitative performance.

VIII. TO GO FURTHER

Despite the significant progress made in our river and road boundary detection project, some images still exhibit poor predictions, indicating room for further improvement. To enhance the accuracy and reliability of our model, several avenues can be explored in future work.

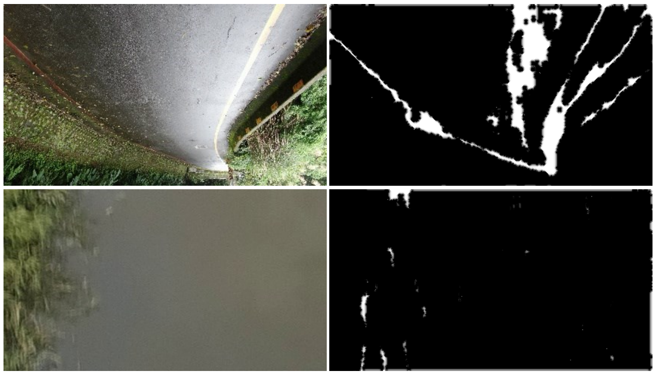


Fig. 26. Example of bad Predictions

First, we could experiment with different neural network architectures. While U-Net has proven effective, other models like ResNet-based U-Nets, SegNet, or more advanced architectures like DeepLab or Mask R-CNN might give better performance, especially in capturing complex features and boundaries.

These models could provide more robust predictions, particularly for challenging images where current methods fall short.

Additionally, refining our loss function could lead to better results. Custom loss functions that better handle class imbalance, such as focal loss or dice coefficient loss, could improve model training by giving more weight to the minority class (white pixels).

This approach could enhance the model's sensitivity to boundaries and improve overall accuracy.

Another area for improvement is data augmentation and preprocessing techniques. Implementing more sophisticated data augmentation strategies, such as random rotations, flips, or elastic deformations, could help the model generalize better by exposing it to a wider variety of scenarios.

Furthermore, advanced preprocessing methods like contrast enhancement or adaptive thresholding might improve the quality of input images, leading to better predictions.

However, our efforts are currently constrained by the performance limitations of our computational resources. Training deep neural networks like U-Net requires substantial processing power and memory, and our current setup is relatively slow, even for moderately complex models. To overcome this, exploring cloud-based solutions or using more powerful hardware could significantly speed up training and allow us to experiment with more complex models and larger datasets.

In conclusion, while our current model and post-processing techniques have achieved promising results, there is considerable potential for further improvement.

By exploring different neural network architectures, refining loss functions, enhancing data augmentation and preprocessing techniques, and addressing computational limitations, we can strive to achieve even more accurate and reliable boundary detection in future iterations of this project.

IX. CONCLUSION

In conclusion, our project aimed to develop an effective river and road boundary detection system using deep learning techniques. Through rigorous experimentation on the model selection and the post treatments methods, we have made significant progress in achieving this goal.

Looking ahead, there are several avenues for future improvement, including exploring alternative neural network architectures, refining loss functions, enhancing data augmentation and preprocessing techniques, and addressing computational limitations.

By addressing these challenges and continuing to iterate on our methodology, we can strive to develop an even more accurate and reliable boundary detection system.

In summary, our project represents a significant step forward in the field of river and road boundary detection, with potential applications ranging from urban planning and environmental monitoring to navigation and infrastructure development.

With continued research and development, we aim to further refine our approach and contribute to advancements in this critical area of study.

X. REFERENCES

[1] Image-to-Image Translation with Conditional Adversarial Networks. (s. d.). IEEE Xplore. https://ieeexplore.ieee.org/document/8100115

[2] Papers with Code - Attention U-Net: Learning Where to Look for the Pancreas. (s. d.). The latest in Machine Learning | Papers With Code, https://paperswithcode.com/paper/attention-u-net-learning-where-to-look-for

[3] DeepUNet: A Deep Fully Convolutional Network for Pixel-Level Sea-Land Segmentation. (s. d.). IEEE Xplore, https://ieeexplore.ieee.org/document/8370071

[4] Python OpenCV - Canny() Function - GeeksforGeeks. (s. d.). GeeksforGeeks, https://www.geeksforgeeks.org/python-opencv-canny-function/

[5] OpenCV: Hough Line Transform. (s. d.). OpenCV documentation index, https://docs.opencv.org/3.4/d9/db0/tutorial_hough_lines.html

[6] 3.4. Metrics and scoring: quantifying the quality of predictions. (s. d.). scikit-learn, https://scikit-learn.org/stable/modules/model_evaluation.html