

# The paper (Ref: Padam\_Hoque\_TR2025\_2) is under Review

## **RuttingNet: Toward Robust Semantic Segmentation in Levee Infrastructure Monitoring: Enhancing Accuracy with High-Fidelity Synthetic Data and Ensemble Learning**

Padam Jung Thapa, Md Tamjidul Hoque

{pthapa, thoque}@uno.edu

Computer Science, University of New Orleans, Louisiana, USA.

## Abstract

Levees serve as critical flood protection structures, but failures due to inadequate maintenance and extreme water pressures have led to devastating events such as Hurricane Katrina. Manual inspections are slow, labor-intensive, and prone to human error, necessitating the development of automated solutions. This study proposes an AI-driven framework for levee inspection utilizing deep learning-based semantic segmentation to detect rutting and enhance the identification of sand boils. To address dataset limitations, high-fidelity synthetic images are generated using DreamBooth for fine-tuning, while ControlNet adds structural constraints to enhance realism and consistency. A semi-automatic convex hull annotation technique enhances labeling efficiency, and ensemble learning strategies further improve segmentation accuracy. The system integrates real-time inference capabilities within a web-based platform, enabling rapid and precise identification of defects. By combining deep learning, synthetic data augmentation, and real-time deployment, this research presents a scalable, innovative solution for automated levee monitoring, addressing key challenges in flood risk management and infrastructure resilience.

Keywords: Levee Inspection, Deep Learning, Semantic Segmentation, Synthetic Data, Real-time Monitoring, Generative AI.

## 1. Introduction

Earthen levees are critical for protecting communities and infrastructure from catastrophic flooding, yet their structural integrity is threatened by defects like rutting and sand boils. The devastating failure of levees during Hurricane Katrina highlighted the severe consequences of inadequate monitoring. Traditional inspection methods are manual, slow, and error-prone, making them inefficient for the vast levee networks that safeguard millions of people and billions of dollars

in assets [1]. This necessitates a shift towards automated, scalable, and precise monitoring solutions.

This research aims to detect and segment levee ruts and sand boils using an enhanced encoder-decoder architecture with transfer learning, leveraging a pre-trained model as a feature extractor to improve efficiency and minimize reliance on large training datasets. A primary challenge in this domain is the scarcity of high-quality, annotated data for training robust models. Our approach is built on a foundation of transfer learning [9], leveraging powerful, pre-trained models to avoid the immense computational cost and data requirements of training from scratch. By fine-tuning a pre-trained backbone, we adapt general visual representations for the specialized task of levee inspection, significantly reducing training time and improving model adaptability. To further enhance robustness and overcome data limitations, we leverage advanced generative AI, specifically fine-tuning text-to-image diffusion models with DreamBooth to generate high-fidelity synthetic defect images. Furthermore, ControlNet is employed to impose structural constraints on the generative process, ensuring the realism and consistency of the synthetic data.

To complete our end-to-end system, we introduce a semi-automatic annotation pipeline using convex hull techniques to efficiently label the generated data, thereby reducing hours of manual effort. Segmentation accuracy is further improved using ensemble learning, which aggregates predictions from multiple specialized U-Net variants to produce more reliable and robust results. Finally, to ensure this research translates into a practical tool, we have developed and deployed a web-based application using Streamlit. This interactive platform supports both image and video overlay for real-time defect visualization, enabling rapid assessment and proactive flood risk management for decision-makers. By integrating transfer learning, generative data augmentation, and a real-time deployment, this work presents a scalable, data-driven solution for levee monitoring.

Our methodology enhances model performance through several key innovations. We introduce a semi-automatic convex hull annotation technique to improve labeling efficiency for the generated data. To maximize segmentation accuracy, we employ an ensemble of specialized U-Net-based architectures [7], including a novel RuttingNet, which aggregates predictions to produce more robust and reliable results. The entire system is integrated into a real-time, web-based platform, providing an accessible tool for rapid and precise defect identification. By combining state-of-the-art deep learning, generative data augmentation, and ensemble strategies, this research presents a scalable and innovative solution that significantly advances automated levee monitoring and strengthens flood risk management.

## 2. Background

Levee defect detection has evolved from manual measurements and classical computer vision to deep learning-based approaches. Early methods, including edge detectors, wavelet transforms, and thresholding techniques like Otsu’s method or Gabor and Canny filters, struggled with lighting variations, shadows, and surface textures, often misidentifying stains or missing subtle anomalies [50]. Traditional machine learning models, such as SVMs trained on handcrafted features, offered slight improvements but remained limited by the quality of the features. Today, deep neural networks, particularly encoder–decoder architectures like U-Net and ResNet, have

revolutionized levee defect detection by enabling precise object-level detection and pixel-level segmentation. Additionally, object detection frameworks such as YOLOv3 and YOLOX-s have shown robust performance on rut detection. Yet their bounding-box outputs can fail to capture the exact shape and severity of defects, underscoring the need for segmentation-based models when higher geometric fidelity is required [51]. Recent work on road-based rutting demonstrated the viability of segmentation methods like PSPNet and DeepLabv3+, achieving around 53–55% IoU, highlighting both the feasibility and challenges of pixel-level rut detection [52]. With transfer learning and fine-tuned models, these systems achieve greater accuracy and consistency, overcoming the limitations of earlier rule-based methods. Below, we discuss key models, techniques, and recent advances in segmentation tailored to infrastructure defects such as rutting and sand boils.

## 2.1 Latest Levee Defect Detection Works

Levee systems can exhibit various structural vulnerabilities—such as cracks, seepages, and sand boils—when subjected to high water pressure, as illustrated in Figure 2.1. Within the context of levee inspection, Panta et al. [26] introduced IterLUNet, a lightweight U-Net derivative that uses iterative skip connections and multi-scale feature fusion, resulting in improved IoU scores for levee crack segmentation. Kuchi et al. [27] further addressed sand boil detection by combining synthetic datasets with a machine learning framework to mitigate class imbalance. Continuing in this domain, Panta et al. [28] developed SandBoilNet—a fully convolutional network that leverages transfer learning for sand boil segmentation—and later extended their work to seepage segmentation [29], effectively handling the complexities of real-world images. Although these efforts demonstrate the success of targeted deep learning architectures for levee-related defects, they often focus on specific fault types, such as cracks, sand boils, or seepages.

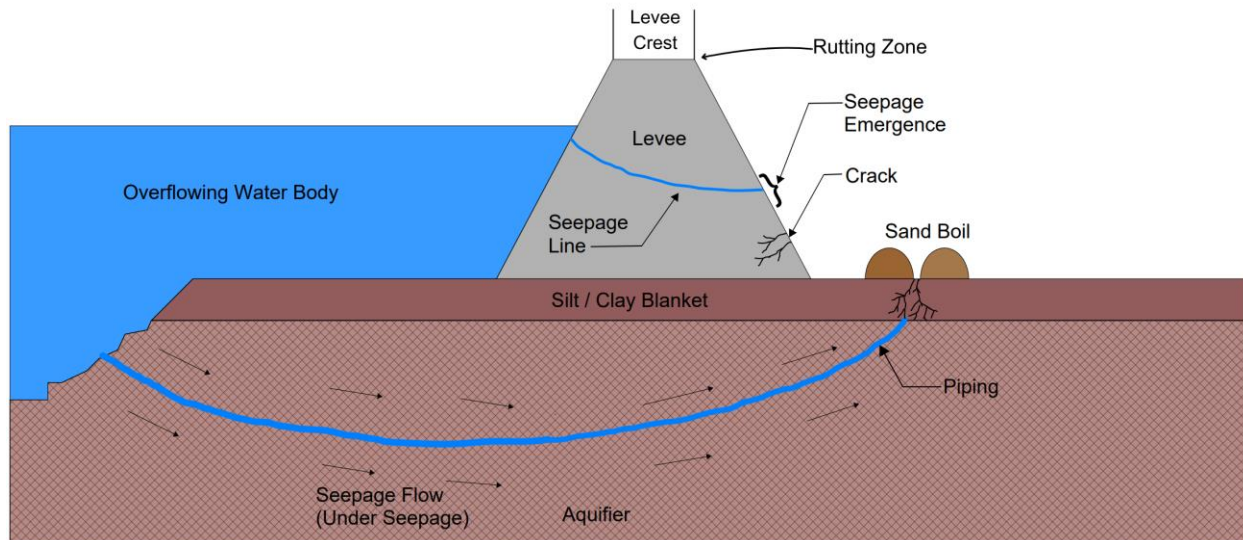
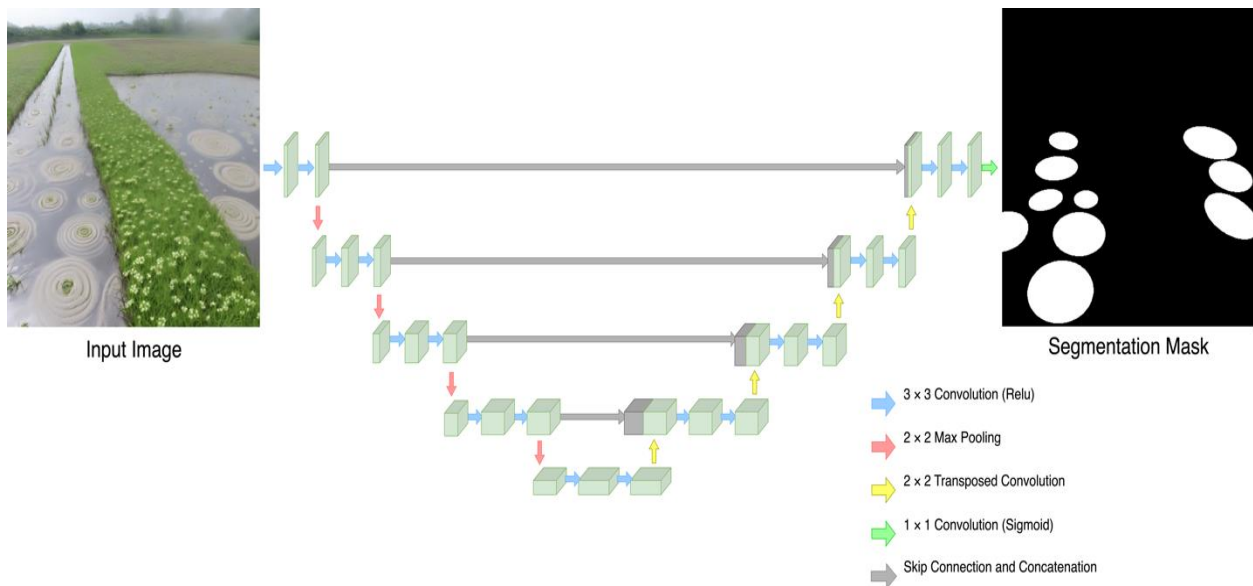


Figure 2.1: Cross-sectional illustration of a levee system highlighting rutting formation zone, cracks, seepage, and sand boil. The high-water level forces water through a permeable sand aquifer—leading to under-seepage and, ultimately, the emergence of sand boils at the surface.

In broadening this scope, Alshawi et al. [30] investigated imbalance-aware culvert-sewer defect segmentation using an Enhanced Feature Pyramid Network (E-FPN) with sparsely connected blocks and depth-wise separable convolutions for enhanced feature extraction. Concurrently, Alshawi et al. [31] introduced a depth-wise separable U-Net with multiscale filters for sinkhole detection, highlighting how thoughtful architectural choices can simultaneously reduce model complexity and enhance segmentation accuracy. These studies collectively highlight the importance of domain-focused approaches for robust infrastructure inspection, while also underscoring the need for more generalized solutions capable of addressing a wider array of levee and infrastructure defects.

## 2.2 Semantic Segmentation

Semantic segmentation is a sophisticated computer vision technique that assigns a class label to each pixel in an image, effectively partitioning the scene into meaningful regions [16]. Unlike image classification, which predicts a single label for an entire image, or object detection, which locates discrete objects via bounding boxes, semantic segmentation produces a dense, pixel-wise classification map for comprehensive scene interpretation [17]. By leveraging Convolutional Neural Networks (CNNs), this approach enables systems to learn feature representations that capture both content and context at a granular level. As a result, semantic segmentation has demonstrated high accuracy in autonomous driving [18], medical imaging [19], and remote sensing [20].



**Figure 2.2:** U-Net architecture for image segmentation. Each  $3 \times 3$  CNN layer includes convolution, batch norm, and ReLU. Downsampling uses max pooling; upsampling uses transposed convolutions. Skip connections fuse encoder and decoder features. A final  $1 \times 1$  convolution with sigmoid yields the segmentation mask.

Modern segmentation architectures frequently build upon Fully Convolutional Networks (FCNs) [21], replacing fully connected layers with convolutions to generate output maps that align spatially with the input. Notable encoder-decoder variants include the Pyramid Scene Parsing Network (PSPNet) [22], which integrates a pre-trained ResNet101 [23] with a pyramid pooling module for multi-scale global context, and SegNet [24], which reuses pooling indices from the encoder to guide the decoder’s upsampling. The U-Net family is especially recognized for its skip connections that preserve fine-grained detail while integrating broader contextual information (see Figure 2.2 for the base architecture). Building on U-Net’s foundations, MultiResUNet [13] incorporates multi-scale processing with residual connections to enhance feature representation, U-Net++ [14] adopts nested and dense skip connections for deeper supervision, and Attention U-Net [15] employs attention gates to emphasize salient regions. Additionally, DeepLabv2/v3 [25] utilizes Atrous Spatial Pyramid Pooling (ASPP) to capture multi-scale context with reduced computational overhead.

Collectively, these encoder-decoder frameworks excel in scenarios that demand both high-level semantic understanding and fine-grained spatial detail, making them integral to a range of applications involving irregular objects or cluttered backgrounds.

## 2.3 Generative Models

Generative models aim to learn the underlying distribution of a dataset, enabling the synthesis of new, realistic samples from this learned representation. Early approaches to generative modeling include Restricted Boltzmann Machines (RBMs) and Deep Belief Networks (DBNs). However, more recent methods, such as Variational Autoencoders (VAEs) [32] and Generative Adversarial Networks (GANs) [33], have significantly advanced the field. VAEs operate by encoding input data into a latent space and then decoding from that space to reconstruct or generate new samples. This latent-space representation enables tasks such as controlled data generation and interpolation between classes. GANs, on the other hand, employ a game-theoretic framework with two competing networks—a generator and a discriminator. The generator attempts to produce realistic samples that can deceive the discriminator, while the discriminator learns to distinguish between generated samples and real ones. This adversarial process often leads to highly plausible synthetic data once both networks reach equilibrium.

The triangle diagram in Figure 2.3 illustrates the trade-offs between three key objectives in generative modeling: high-quality samples, fast sampling, and mode coverage/diversity. Generative Adversarial Networks (GANs) sit between high-quality outputs and fast sampling, making them ideal for producing realistic images quickly. However, they often suffer from mode collapse, where the model fails to generate the full diversity of the data distribution. On the other hand, Variational Autoencoders (VAEs) and Normalizing Flows prioritize fast sampling and diverse mode coverage, effectively capturing a wide range of variations in the data. Their main drawback lies in producing less visually convincing samples compared to GANs or diffusion models.

Denosing Diffusion Models (DDMs) strike a different balance by excelling in both sample quality and mode diversity. They generate high-fidelity and diverse outputs but rely on a slow, iterative denosing process, which makes sampling significantly slower. The figure emphasizes

that no current model perfectly satisfies all three objectives; each model type lies closer to two corners of the triangle, highlighting the trade-offs involved. Understanding these placements helps in selecting the appropriate model for a given application based on the desired balance of quality, speed, and diversity.

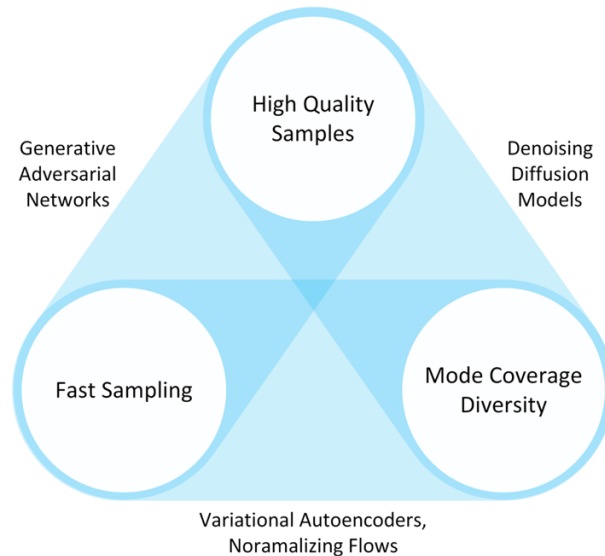


Figure 2.3: Comparisons of Generative AI Models Based on Sample Quality, Sampling Speed, and Mode Coverage Diversity

These generative approaches offer notable advantages in domains where data scarcity or imbalance is prevalent. For instance, conditional GANs [34] can synthesize class-specific training samples, helping address uneven class distributions or limited annotated images. In remote sensing, they can generate additional training examples for rare phenomena, and in medical imaging, they can augment datasets of uncommon pathologies without compromising patient privacy. Recent variants of GANs and VAEs also incorporate attention mechanisms, multi-scale feature fusion, or self-supervised learning components to bolster the quality and diversity of generated outputs [35,36].

Integrating generative models with semantic segmentation pipelines has become increasingly common. By producing synthetic images that closely mimic real-world conditions, these models can expand the training set for segmentation networks and improve robustness in scenarios where data collection is difficult, expensive, or hazardous. In the context of levee inspection, generative models can help create representative samples of rare fault instances—be they cracks, sand boils, seepages, rutting, encroachments, or other defects—thereby mitigating class imbalance and enhancing the training of segmentation architectures. As a result, generative models serve as a crucial tool for both augmenting existing datasets and exploring new strategies to enhance the performance, generalizability, and resilience of deep segmentation frameworks.

Over the past few years, StyleGAN and its improved variant StyleGAN2-ADA [37,38] have demonstrated remarkable capabilities in synthesizing high-quality images, including face images and other complex objects. However, in certain niche domains—such as levee fault

imagery—data availability can be severely constrained. Under these limited data conditions, StyleGAN-based models may struggle to produce diverse synthetic samples, thereby restricting their utility for tasks such as data augmentation in fault detection or segmentation.

To overcome these limitations, researchers have increasingly turned to diffusion-based models, which iteratively denoise random noise into coherent images. Methods like Stable Diffusion [39] and its extensions incorporate guidance strategies—ranging from text prompts to structural inputs—that can yield broader variation and finer control over the generated content. Two notable diffusion-based techniques are DreamBooth [11] and ControlNet [12]. DreamBooth refines a base diffusion model using a small number of domain-specific examples, enabling more precise, subject-driven generation. ControlNet extends diffusion architectures by incorporating control mechanisms (e.g., edge maps, pose estimations) that guide the generation process, enabling the maintenance of structural consistency or the achievement of a desired image-to-image transformation.

In the context of levee inspection, transitioning from traditional GAN-based strategies to diffusion-based approaches can prove advantageous. By incorporating textual or structural prompts, diffusion models can generate a more diverse range of synthetic images that better emulate real-world defects and environmental variability. This enhanced variety is crucial for tackling data imbalance and scarcity, as it expands training sets for downstream segmentation tasks. Consequently, advanced generative approaches not only bolster model generalization but also pave the way for more robust, domain-focused solutions in infrastructure and remote sensing applications.

## 2.4 Rutting Detection

Rutting refers to the formation of deep grooves or depressions in soil or a levee surface due to erosion, vehicle movement, or water flow. While typically associated with pavements, rutting in levee infrastructure can undermine the structural integrity of flood defenses by creating low-lying areas that accumulate water. Excessive rutting may increase the likelihood of seepage or breaches during high-water events, thereby amplifying the risk of levee failure.

Traditional rutting assessment methods rely on manual measurements, such as straightedges or level rods, which can be time-consuming and prone to human error. To improve efficiency and accuracy, modern inspection approaches often utilize 3D laser scanning or LiDAR to capture high-resolution surface profiles [40] and then apply machine learning algorithms to distinguish genuine deformations from benign surface variations. Notably, LiDAR sensors mounted on mobile platforms—including forestry machines—have shown promise for large-scale rut depth measurements, achieving unbiased estimates with minimal error [41]. Beyond these established techniques, several innovative frameworks have recently emerged: a metaheuristic-optimized machine learning model that employs Gabor filters and discrete cosine transforms for image-based rut detection [42], a cost-effective measurement system combining a linear laser and high-speed camera [43], and a tile-based LiDAR method that uses road surface fitting for comprehensive rut depth assessments [44]. Ground Penetrating Radar (GPR) further enhances these surface-based evaluations by probing subsurface conditions—such as moisture accumulation, voids, or weakened substrates—that contribute to permanent deformations. GPR transmits electromagnetic waves into the levee layers, analyzing reflections from interfaces with

varying dielectric properties, thereby offering more profound insight into potential causes of rutting and guiding proactive maintenance interventions [45]. By integrating these advanced sensing and modeling approaches, agencies can anticipate rut development, minimize seepage risks, and fortify levee systems against extreme flood events. This data-driven strategy reduces maintenance costs, enhances flood protection, and extends the service life of the levee. However, to the best of our knowledge, no prior work has employed pixel-level semantic segmentation specifically tailored for levee rutting. This study aims to fill that gap by offering a novel segmentation-based framework, designed to provide high-resolution detection ultimately advancing the state of the art in levee infrastructure maintenance.

## 2.5 Sand Boil Detection

In addition to advancing rutting detection, this study also focuses on refining sand boil detection by generating synthetic datasets [48] and improving annotation strategies. Sand boils are surface manifestations of internal erosion within levee systems, typically forming when water under high hydraulic pressure infiltrates permeable subsurface layers, carrying fine sediments upward. This process results in dome- or cone-shaped mounds at or near the surface, as illustrated in Figure 2.1. These anomalies frequently occur during flood events or prolonged high-water conditions and are early indicators of piping—a critical subsurface erosion mechanism that, if left unmitigated, can lead to levee instability or catastrophic failure. Prompt and precise detection of these formations is therefore vital for early intervention and levee resilience.

Traditional detection methods for sand boils have largely depended on manual visual inspection during high-water events, where field personnel survey levee surfaces to identify sediment eruptions. While operationally straightforward, such approaches are time-consuming, labor-intensive, and prone to human error, particularly in large-scale flood-prone areas. Moreover, poor lighting, environmental variability, and the presence of similar surface anomalies can further complicate reliable detection.

In response to the limitations of manual inspection, researchers such as James V., et al. [49], have explored remote sensing techniques, including Synthetic Aperture Radar (SAR), for monitoring subsurface water movement and detecting anomalies indicative of sand boil formation. SAR’s ability to operate in low visibility and penetrate soil layers makes it a promising candidate for early detection. However, due to environmental interference and resolution constraints, SAR often falls short in accurately identifying the small, spatially confined features typical of early-stage sand boils.

In parallel, early computational techniques employed classical machine learning approaches, using handcrafted features such as shape descriptors, edge gradients, and intensity-based thresholds, to detect sand boil-like structures in levee imagery. While effective in controlled environments, these methods struggled to generalize across real-world conditions due to very limited datasets and variations in background textures, soil color, and lighting. More recently, deep learning-based models have shown promise in improving detection accuracy. For example, Kuchi et al. [27] utilized convolutional neural networks (CNNs) trained on manually annotated datasets to identify sand boils from aerial imagery, demonstrating improvements over traditional classifiers, particularly in complex field scenarios. However, their work did not utilize synthetic data, which limits its applicability in highly imbalanced or data-scarce conditions. Building on these



foundations, Panta et al. [28] introduced a custom semantic segmentation architecture designed for pixel-level sand boil delineation—which outperformed prior shallow models in both accuracy and spatial consistency.

Despite these advances, challenges remain. The limited availability of high-quality, pixel-level annotated sand boil datasets restricts the use of existing deep learning models. This research addresses that gap by introducing synthetically generated sand boil datasets and novel annotation pipelines, ultimately improving generalization and segmentation precision across real-world levee imagery.

## Data and Methodology

### 3.1 Introduction

Levees are vital in mitigating flood risks, but they remain susceptible to various forms of deterioration over time. Among these, the most pressing concerns are rutting—linear depressions or grooves in the levee’s surface—and sand boils—funnel-shaped formations caused by subsurface erosion. Both phenomena can compromise the structural integrity of flood-control systems if left unaddressed. Conventional inspection methods often rely on manual surveys and visual checks, which can be impractical for large-scale networks of levees. Current USACE inspection protocols classify rut severity based on depth thresholds, categorizing depressions exceeding 6 inches as unacceptable (U-rated) due to their demonstrated hydraulic risks [1]. Maintenance guidelines mandate immediate remediation of U-rated features, while shallower (<6") depressions receive minimally acceptable (M) ratings requiring monitoring and scheduled repairs [46]. To address these challenges, this chapter outlines a comprehensive data and methodology framework that leverages deep learning for automated fault detection. We focus primarily on rutting—assembling a real corpus of real images, creating synthetic examples to mitigate data limitations, and applying extensive augmentations to enhance model robustness. In parallel, we refine existing sand boil detection approaches by generating additional synthetic sand boil images, further expanding the dataset and enhancing segmentation accuracy for these critical levee datasets.

### 3.2 Rutting Dataset

Ruts and depressions represent critical maintenance concerns for levee systems, developing through multiple mechanisms including vehicular compaction from patrol or maintenance traffic, differential settlement of embankment materials, or insufficient crown slope gradients that impede drainage. The real-world rutting dataset used for this study was self-collected from public online sources (e.g., Google) and, in part, from the U.S. Army Corps of Engineers (USACE) manual by leveraging official documentation where rutting in levee contexts is occasionally illustrated [1]. From an initial pool of inspection photographs that served as source material for synthetic image generation, 20 real images containing clearly identifiable rutting features were selected through systematic curation. Image selection followed a rigorous protocol prioritizing unambiguous visibility of rutting manifestations across critical levee infrastructure components. The dataset encompasses diverse levee morphology through images capturing three

structural domains: crest zones subjected to recurring maintenance vehicle traffic, side slope regions vulnerable to drainage-related erosion, and transitional areas connecting these features. As shown in Figure 3.1, real-world rutting examples vary across different levee sections. Since many real levee images available online or in manuals often show wide aspects rather than close-up defect details, each chosen image was verified to contain a distinct rutting region. This filtering process aimed to ensure a high signal-to-noise ratio, where rutting is prominent enough for precise annotation.



(a)



(b)



(c)



(d)

Figure 3.1: Representative real-world examples of rutting across different levee sections, including crest zones (a, c, d) and side slopes (b), illustrating variations in rut severity, morphology, and environmental context.

## 3.2 Sand Boil Dataset

Sand boils are among the most critical defects captured by the U.S. Army Corps of Engineers (USACE) during their routine level inspections. Field inspectors systematically drive along the levee crest and adjacent areas, using cellphones or high-resolution cameras to capture images of any potential abnormalities they encounter. These photographs are uploaded to a centralized database, which contains over 4,000 levee-related defect images, including instances of cracks, seepages, and sand boils. For this study, a curated subset of approximately 300 sand boil images was selected from the larger USACE repository, based on expert recommendations. The goal was to ensure each image chosen displayed clearly discernible sand boil characteristics in a variety of environmental contexts, ranging from grassy and muddy terrains to concrete and pebble-strewn surfaces. This approach maximizes the dataset’s diversity, offering robust coverage of color, texture, and lighting conditions inherent to real-world levee inspections.

Once the images were filtered, each sand boil example was carefully examined to confirm it depicted either a single dominant boil or multiple closely spaced boils requiring segmentation. These filtration steps aimed to eliminate ambiguous or low-quality images that could introduce unwanted noise into the dataset. Consequently, the curated sand boil set exhibits a representative spectrum of shapes and sizes, which are critical for developing reliable segmentation models. This rigorous selection process aligns with standard practices for semantic segmentation tasks, where data quality and diversity are key factors in achieving high performance. By integrating expert

judgment with methodical reviews of USACE field imagery, the sand boil dataset forms a solid foundation for training and evaluating computer vision algorithms dedicated to levee health monitoring.



(a)



(b)



(c)



(d)

Figure 3.2: Real-world examples of sand boils captured across diverse levee environments, illustrating variability in appearance, terrain, and contextual challenges—ranging from grassy (a), muddy agricultural (b),

and sediment-rich (c) settings to controlled sandbag-surrounded conditions (d) commonly encountered during field inspections.

Despite their typically round or elliptical appearance, sand boils pose several detection challenges. They can be small and intricate, making them difficult to distinguish from the surroundings, especially in wet grassland areas, as shown in Figure 3.2(a). Additionally, various forms of noise affect sand boil images, including variations in texture and color between the boils and their surrounding areas (evident in Figures 3.2(b) and 3.2(c)), hampering identification based solely on visual cues. In some cases, sand boils may be surrounded by sandbags to prevent further enlargement, as illustrated in Figure 3.2(d).

In this investigation, existing sand boil images from levee inspection archives were supplemented with synthetic samples to mitigate real-world data limitations. This approach introduced variations in texture, moisture levels, and surrounding terrain, which helped the segmentation model adapt to the noise, color shifts, and shape irregularities observed in real inspections. By integrating refined sand boil detection into our broader methodology, the framework aims to strengthen overall levee fault monitoring, reducing the chance of embankment failure during floods and minimizing disaster risk.

### 3.3 Generative Approaches for Synthetic Data Augmentation

This section introduces the motivation behind utilizing generative modeling techniques to address the critical challenge of limited annotated data in levee defect segmentation. Manual data collection for geotechnical anomalies such as rutting and sand boils is often constrained by environmental conditions, safety concerns, and the rarity of certain defect types—resulting in small, imbalanced datasets that hinder the performance of deep learning models. To overcome this bottleneck, synthetic data generation using generative models offers a scalable and effective solution. This study investigates two major classes of generative frameworks—Generative Adversarial Networks (GANs) and Diffusion-based Models—each offering distinct advantages in image realism, diversity, and controllability. While GANs have been widely used for high-fidelity image synthesis, they are prone to challenges such as mode collapse and training instability, especially under low-data regimes. In contrast, diffusion models provide greater robustness and flexibility, particularly when guided through conditioning techniques. The following subsections explore these two paradigms in detail, highlighting their roles in augmenting levee inspection datasets and improving segmentation performance.

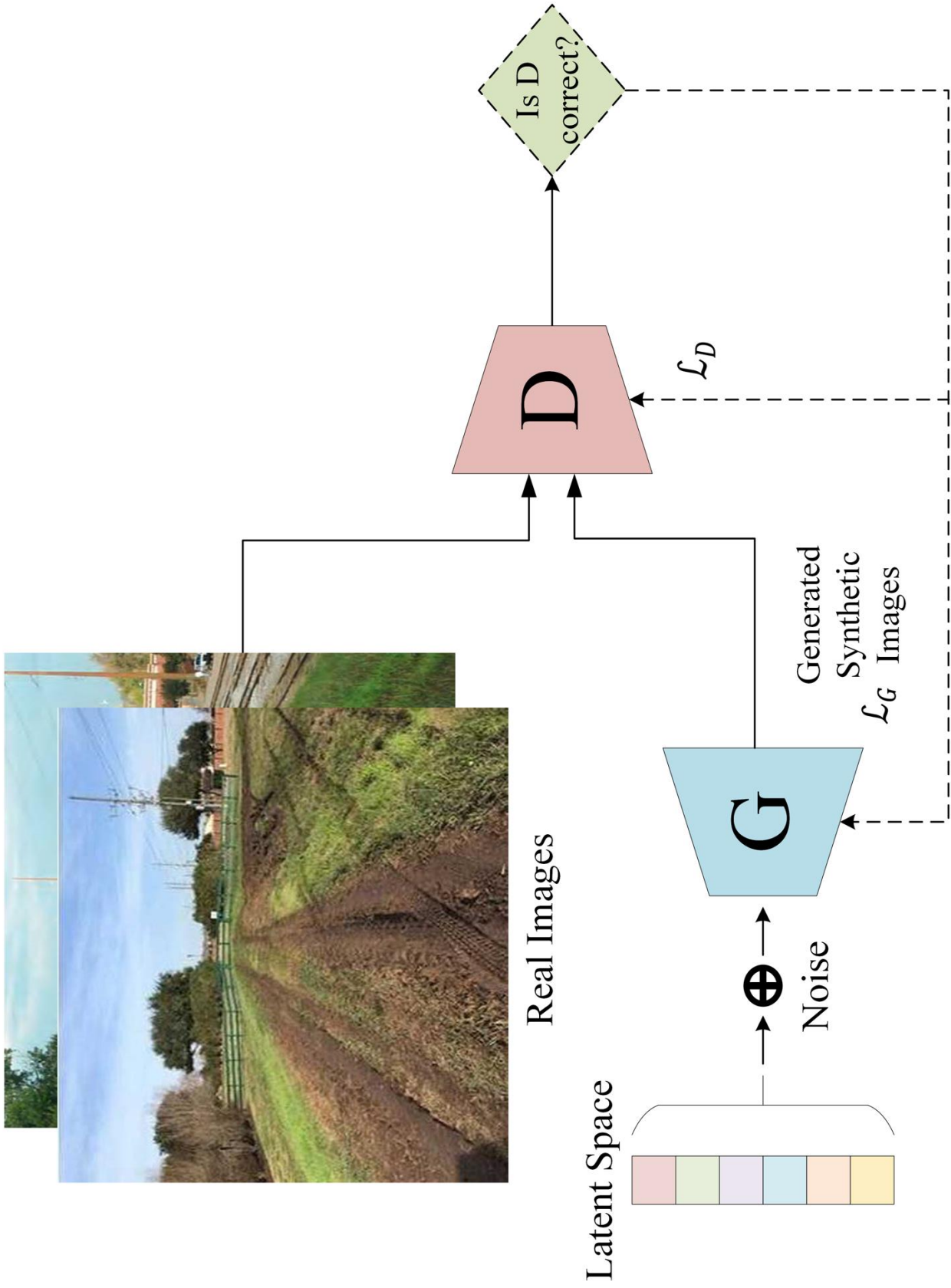
#### 3.3.1 GAN-Based Approaches

##### A. Early Experiments with StyleGAN2-ADA

This study initially explored generative adversarial networks (GANs), specifically the StyleGAN family, for synthetic levee fault image augmentation. GANs, first introduced by Goodfellow et al. [33], are generative models composed of two competing neural networks: the generator (G), which creates synthetic data, and the discriminator (D), which assesses whether a given sample is real or synthetic. Through iterative adversarial training, these two networks improve each other simultaneously. The generator tries to fool the discriminator, while the discriminator tries to correctly identify real versus fake inputs, as shown in Figure 3.3. This adversarial dynamic is formalized as a min-max optimization game as shown in Equation 3.1:

$$\min_G \max_D V(D, G) = E_{x \sim p_{\text{data}}} [\log D(x)] + E_{z \sim p_z} [\log (1 - D(G(z)))] \quad (3.1)$$

In this formulation, G represents the generator network and D represents the discriminator network. The term  $x \sim p_{\text{data}}$  denotes samples drawn from the real data distribution, while  $z \sim p_z$  denotes latent noise vectors sampled from a predefined prior distribution (typically Gaussian). The function  $D(x)$  estimates the probability that input  $x$  is real, and  $G(z)$  is the synthetic image generated by passing  $z$  through the generator. The objective is for D to correctly classify real and fake images, while G tries to generate images that maximize the discriminator's classification error, forming a minmax game. This process ideally results in highly realistic synthetic images that are indistinguishable from real examples.



**Figure 3.3:** Overview of the GAN architecture for synthetic image generation. A latent noise vector is input to the generator  $G$  to produce synthetic images, which, along with real images, are evaluated by the discriminator  $D$  to distinguish authenticity, optimized via losses  $\mathcal{L}_G$  and  $\mathcal{L}_D$ . Training proceeds adversarially until  $G$  generates images

StyleGAN2, developed by Karras et al. [38], is a state-of-the-art GAN variant known for its distinctive style-based architecture. Unlike traditional GANs, StyleGAN2 transforms a latent noise vector through a mapping network into an intermediate latent representation, which is then used to modulate each convolutional layer of the generator independently—a technique known as Adaptive Instance Normalization (AdaIN), allowing for explicit control over high-level attributes and fine-grained visual details. To address stability and artifact issues in earlier versions, StyleGAN2 introduced two significant improvements: the use of weight demodulation in place of AdaIN normalization to eliminate “blob” or “water droplet” artifacts, and a revised network architecture using multi-scale skip connections and residual connections (inspired by MS-GAN) to enhance training convergence and spatial consistency. These enhancements allow StyleGAN2 to generate more stable and high-fidelity outputs, even at high resolutions.

A significant limitation of GANs in general—and StyleGAN variants specifically—is the necessity for extensive training data to prevent discriminator overfitting and the resultant training instability. In practice, acquiring a large dataset is challenging, particularly in specialized domains such as levee fault imaging. Traditional data augmentation methods, such as rotation, cropping, and color transformations, partially mitigate data scarcity. However, they carry risks, notably that artifacts from these augmentations can propagate into the generated images—noise introduced to real inputs, for example, can appear explicitly in synthetic images.

To address these challenges, Karras et al. [38] introduced StyleGAN2-ADA (Adaptive Discriminator Augmentation), an extension of the StyleGAN2 architecture equipped with a novel augmentation strategy. Instead of applying augmentation solely to input training images, ADA systematically applies stochastic transformations to all images shown to the discriminator during training. Importantly, ADA dynamically adjusts augmentation strength based on the discriminator's feedback: if the discriminator overfits, augmentation intensity increases, and vice versa. This adaptive augmentation strategy, built upon balanced consistency regularization (bCR), substantially reduces training dataset requirements—by up to 10–20 times—without significantly compromising image fidelity [56].

In our experiments, StyleGAN2-ADA was trained following the official workflow provided by NVIDIA’s implementation. The image dataset was first preprocessed into TFRecord format and resized to a consistent resolution of  $512 \times 512$  pixels to match the model’s architectural requirements. Fixed random seeds were used throughout the training and generation process to ensure reproducibility. The training leveraged mixed precision (FP16) for faster computation and included periodic saving of generated image samples and FID evaluation metrics during training iterations. The generator model was optimized using a non-saturating logistic loss function, while the discriminator used R1 regularization. Despite leveraging high-end hardware NVIDIA A100 80GB GPUs, the training process was compute-intensive and typically required multiple days to converge—particularly due to the limited variability in the dataset.

Ultimately, however, StyleGAN2-ADA did not yield satisfactory results for the levee fault dataset. While the model produced images with relatively realistic features, we consistently observed limited variation among the generated images. This phenomenon, known as mode collapse, occurs when the generator fails to capture the entire data distribution and instead produces a narrow subset of repetitive samples. Mode collapse is common in GANs trained on datasets exhibiting inherently low variability, as limited diversity makes it easier for the generator to settle into generating only a few plausible images that sufficiently fool the discriminator. Additionally, significant training instabilities were observed, including difficulties balancing the generator and discriminator, vanishing gradients, oscillating training dynamics, and high



sensitivity to hyperparameter choices, despite using high-end computational resources. Thus, the model did not enhance dataset diversity as desired, effectively negating the benefits of synthetic data augmentation for our segmentation task.

Evaluation of generated image quality typically involves metrics such as Fréchet Inception Distance (FID) and Kernel Inception Distance (KID). Briefly, FID computes the Wasserstein-2 distance between real and generated image distributions in a pretrained Inception-v3 feature space, where lower values indicate superior image realism and variety. KID similarly measures statistical differences in feature distributions but is more unbiased and reliable, particularly with smaller datasets. Due to notable mode collapse and training instabilities, StyleGAN2-ADA demonstrated unsatisfactory FID and KID scores in our scenario.

Given these limitations, our research transitioned toward more advanced generative modeling techniques. Recent conditional image generation frameworks, specifically diffusion-based models such as DreamBooth and ControlNet, offer improved robustness in scenarios with limited data. These methods leverage extensive, pre-trained visual priors to effectively mitigate the issues of limited variation and mode collapse, thereby supporting a more diverse and high-quality synthetic dataset suitable for enhancing semantic segmentation models.

### 3.3.2 Diffusion-Based Approaches

Considering the limitations encountered with GANs, this subsection transitions to diffusion-based generative models, highlighting their robustness in limited-data scenarios due to enhanced diversity and control. It provides an overview of the general diffusion process and introduces two specialized variants—DreamBooth for fine-tuning and ControlNet for structural guidance—each of which is described in subsequent detailed subsections.

#### A. Overview of the Stable Diffusion Model

While Generative Adversarial Networks (GANs), such as StyleGAN2-ADA, have previously demonstrated potential for synthesizing realistic images, they often encounter issues, including mode collapse, training instabilities, and limited dataset variation—particularly evident in the generation of specialized geotechnical anomalies, like levee defects. To address these shortcomings, recent advancements have shifted toward diffusion-based generative models, which offer a fundamentally different and more stable approach to image synthesis. Unlike GANs, which rely on adversarial training between generator and discriminator networks, diffusion models employ a probabilistic framework that progressively corrupts data with noise and learns to reverse this corruption, thereby recovering the original data distribution.

Diffusion models operate in two phases: a forward diffusion process, where input data is gradually transformed into noise, and a reverse denoising process, where a neural network is trained to reconstruct the original image from noise. This two-step mechanism ensures robust and diverse generation, helping avoid issues like mode collapse that often affect GANs. The effectiveness of this structured noise addition and removal process is visually demonstrated in the diffusion-based generative pipeline, as illustrated in Figure 3.4. The forward process adds Gaussian noise in small increments over a fixed number of timesteps, as defined in Equation 3.2:

$$q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t I) \quad (3.2)$$

Here,  $x_t$  represents the intermediate noisy images at timestep  $t$ , and  $\beta_t$  denotes the noise variance at each timestep. By the end of this diffusion sequence, the data is reduced to pure noise  $x_t$  typically sampled from a standard Gaussian distribution.

The core learning task occurs during the reverse diffusion phase, where a neural network (often implemented as a U-Net) is trained to systematically remove the noise from these corrupted images, gradually reconstructing realistic images from pure noise. This reverse denoising process is defined in Equation 3.3:

$$p_{\theta}(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_{\theta}(x_t, t), \Sigma_{\theta}(x_t, t)) \quad (3.3)$$

In this formulation,  $\mu_{\theta}$  and  $\Sigma_{\theta}$  represent the predicted mean and variance by the neural network parameterized by  $\theta$ . Through iterative denoising, the trained model generates visually coherent images that closely resemble the original training data.

To train the denoising model effectively, it is essential that the learned reverse process  $p_{\theta}(x_{t-1} | x_t)$  closely approximates the true posterior distribution  $q(x_{t-1} | x_t, x_0)$ , which represents the ideal step to reverse noise at each timestep. Since the true posterior is derived using Bayes’ theorem but is intractable to compute directly during inference, diffusion models are trained by minimizing the Kullback-Leibler (KL) divergence between the model’s approximation and the actual posterior. This training objective is formalized in Equation 3.4:

$$\mathcal{L}_{KL} = \text{KL}(q(x_{t-1} | x_t, x_0) \parallel p_{\theta}(x_{t-1} | x_t)) \quad (3.4)$$

Equation 3.4 serves as the core loss function in diffusion training. The KL divergence is a mathematical measure of how one probability distribution diverges from a second, expected distribution. In simpler terms, it quantifies how much information is lost when the model’s predicted distribution  $p_{\theta}$  is used in place of the true distribution  $q$ . A KL divergence of zero would mean the two distributions are identical.

During training, the model learns the parameters of  $p_{\theta}$ —such as the mean and variance of the Gaussian distribution used for denoising—by minimizing this KL divergence using gradient descent. This optimization process iteratively updates the model weights so that its predicted denoising steps more closely approximate the true reverse process.

Rather than reconstructing the original image in a single pass, the model learns a series of small, probabilistically accurate steps to gradually remove noise. This structured and statistically grounded process enables the model to generate outputs that are both realistic and diverse.

During training, the model is optimized using a simplified denoising objective that aims to predict the noise added to a clean image at a given time step. Rather than directly reconstructing the original image  $x_0$ , the network  $\epsilon_{\theta}$  is trained to estimate the noise  $\epsilon$  used in the forward diffusion step. This is more stable and effective, especially when the input is heavily corrupted by noise. The corresponding loss function is expressed in Equation 3.5:

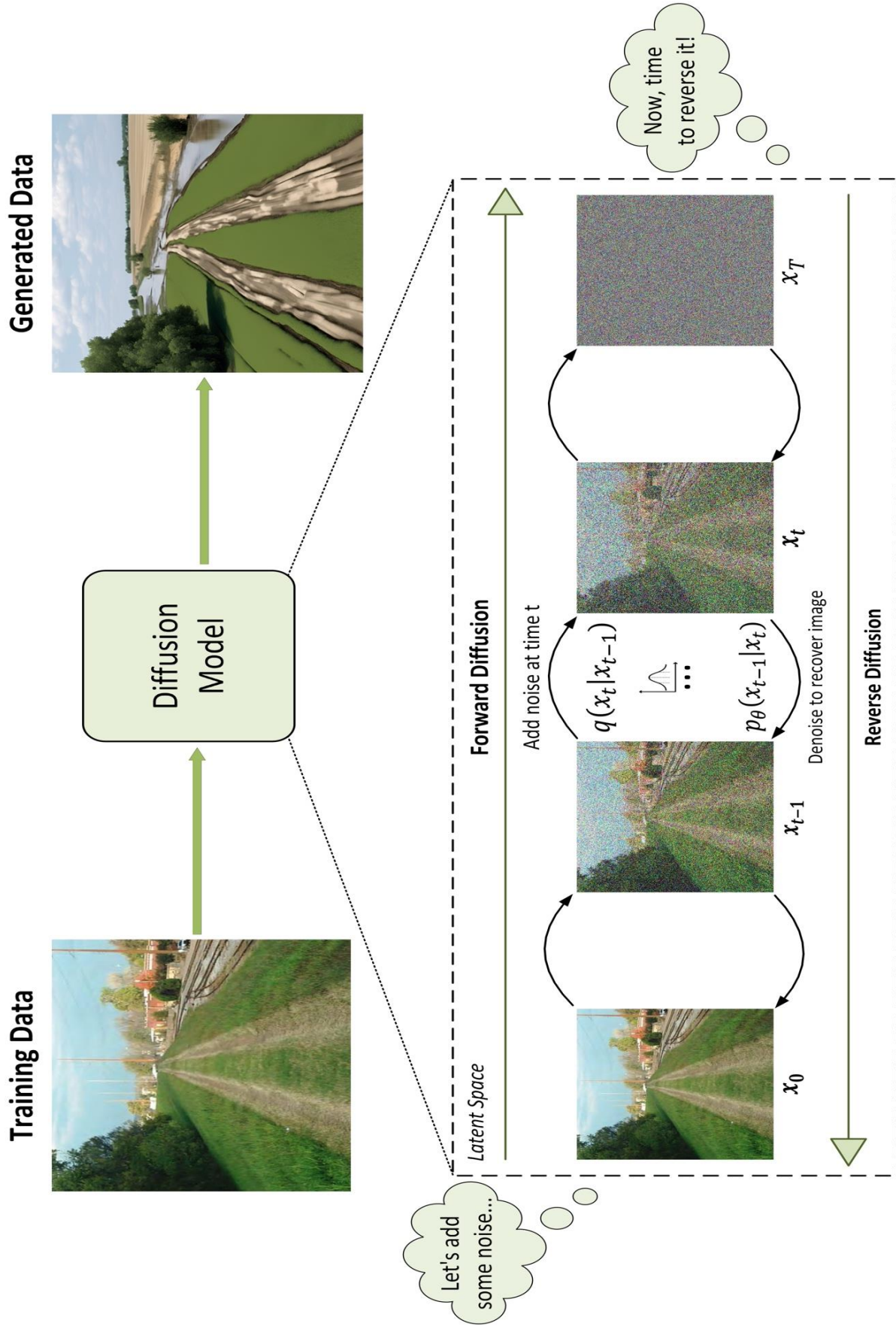
$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t, x_0, \epsilon} [|\epsilon - \epsilon_{\theta}(x_t, t)|^2] \quad (3.5)$$

Here,  $x_t$  is the noisy image at timestep  $t$ , and  $\epsilon_{\theta}$  is the network’s prediction of the noise. This objective function is computationally efficient and empirically effective, and it forms the backbone of training in both standard and latent diffusion models like Stable Diffusion.

To improve computational efficiency, Stable Diffusion operates in a latent space rather than directly on high-resolution pixel images. An encoder  $\mathcal{E}(x)$  compresses images into a lower-dimensional latent space  $z$ , where the diffusion process occurs. The denoised outputs are then decoded back into image space using a decoder  $\mathcal{D}(z)$ , as defined in Equation 3.6:

$$z = \mathcal{E}(x), \quad x' = \mathcal{D}(z) \quad (3.6)$$

This approach—known as Latent Diffusion—greatly reduces the memory footprint and computational cost, allowing for high-resolution image synthesis on modern GPUs without compromising visual quality.



**Figure 3.4:** Schematic illustration of the diffusion-based generative process, showcasing forward (noise addition) and reverse (denoising) diffusion steps, illustrating stable reconstruction from random noise to realistic synthetic images.

Stable Diffusion further enhances these diffusion processes by integrating textual conditioning, enabling highly controllable text-to-image generation. By leveraging latent space embeddings, it translates descriptive prompts into coherent visual representations. Such control over generation significantly surpasses traditional GAN capabilities in terms of stability, realism, and adaptability to specialized tasks like levee defect synthesis. Subsequent sections (DreamBooth and ControlNet) will detail advanced methods built upon this diffusion-based foundation, demonstrating their efficacy in specialized geotechnical contexts.

## B. Dreambooth Fine-Tuning

Automated levee inspection faces a critical challenge: limited training data for rare defects like rutting and sand boils. Training diffusion models from scratch is particularly challenging in such scenarios, as it requires massive datasets, extensive computational resources, and significant training time to achieve stable and realistic image generation. DreamBooth fine-tuning addresses this constraint by leveraging a pre-trained stable diffusion model to learn specific defect characteristics from minimal reference images (typically 10–15 per defect type). This approach leverages transfer learning principles by starting with a model already proficient in understanding general visual concepts—soil textures, vegetation patterns, and environmental conditions—and then guiding it to internalize the distinctive visual signatures of levee-specific anomalies.

The process effectively embeds novel defect representations within the model's extensive latent space, preserving its foundational understanding of natural scenes while developing specialized capabilities for reproducing domain-specific features with high fidelity. This technique enables the creation of diverse, realistic training examples that would otherwise be impossible to collect through traditional field surveys.

Fine-tuning diffusion models for specialized geological features requires a strategic approach. During this process, we provide carefully curated image sets depicting target concepts—whether elongated depressions characteristic of rutting or circular mounds typical of sand boils. These images are paired with two types of textual prompts that serve distinct purposes.

The model learns through both instance prompts and class prompts, each serving a different function. An instance prompt such as “a photo of [identifier] sand boil” connects to the specific new subject being taught. Here, the term “identifier” represents a unique token or special word (often a random string like “sks” or a descriptive term like “geological”) that helps the model distinguish this particular version of a sand boil from others in its training data. Meanwhile, class prompts like “a photo of a sand boil” maintain the link to broader data distributions and general understanding of the concept. This dual-prompt approach allows the model to recognize the specific features of interest while still placing them within the appropriate conceptual category.

During fine-tuning, a technique known as prior preservation penalizes the model from deviating too far from its baseline understanding, creating a crucial balance between capturing unique defect features (specificity) while maintaining overall image realism (generality). This is particularly important when working with geological features, as they must appear natural and consistent with real-world observations despite limited training examples. Moreover, DreamBooth incorporates classifier-free guidance during the generation phase, blending unconditional and conditional noise predictions to improve alignment between the generated images and provided textual prompts. The scale is set up as 7.5 as the standard practice; however, it is adjustable between 5 and 15. This guidance mechanism is mathematically expressed in Equation 3.7:

$$\epsilon_{\text{guided}} = (1 + w) \epsilon_{\theta}(x_t, t, c) - w \epsilon_{\theta}(x_t, t) \quad (3.7)$$

Here,  $w$  represents the guidance scale, which controls prompt adherence strength, allowing generation to be highly aligned with textual descriptions without compromising diversity or realism.

The technical implementation of DreamBooth fine-tuning utilizes conservative parameters due to the significant challenges associated with training diffusion models from scratch, including massive dataset requirements, extensive computational resources, and prolonged training times. Instead, fine-tuning was performed starting from a pretrained Stable Diffusion v1.5 model already proficient in general visual concepts. To effectively specialize the model for generating levee defect images, minimal instance-specific images (typically 10–15 per defect type) were paired with approximately 10–12 times as many generalized class images, thereby preserving broader visual knowledge and preventing overfitting. All images were standardized to a 512×512-pixel resolution for consistent model input. Training involved approximately 80 iterations per instance image (800–3200 total steps), using a low learning rate ( $1e-6$ ) to minimize drastic parameter shifts, coupled with a learning-rate warmup phase (10% of total steps) to ensure training stability. Small batch sizes, typically one image per step, further helped mitigate overfitting. The Adam optimizer with 8-bit precision and mixed-precision training (FP16) was utilized to optimize computational efficiency on an NVIDIA A100 GPU using standard diffusion-model libraries including PyTorch, Diffusers, Accelerate, and xFormers — the latter providing efficient attention mechanisms and memory optimizations for large-scale transformer models.

During inference, the deterministic Denoising Diffusion Implicit Models (DDIM) were specifically chosen over the stochastic Denoising Diffusion Probabilistic Models (DDPM) to significantly reduce the computational load. DDPM sampling requires extensive iterative denoising steps, as defined in Equation 3.8:

$$p_{\theta}(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_{\theta}(x_t, t), \Sigma_{\theta}(x_t, t)) \quad (3.8)$$

where  $x_t$  is the noisy image at timestep  $t$ , and  $\mu_{\theta}$ ,  $\Sigma_{\theta}$  represent the mean and variance predicted by the neural network. Due to its stochastic nature, DDPM sampling typically involves hundreds to thousands of steps, making it computationally demanding.

Conversely, DDIM sampling is deterministic and efficiently approximates the original data distribution using significantly fewer inference steps (25–50), as defined in Equation 3.9:

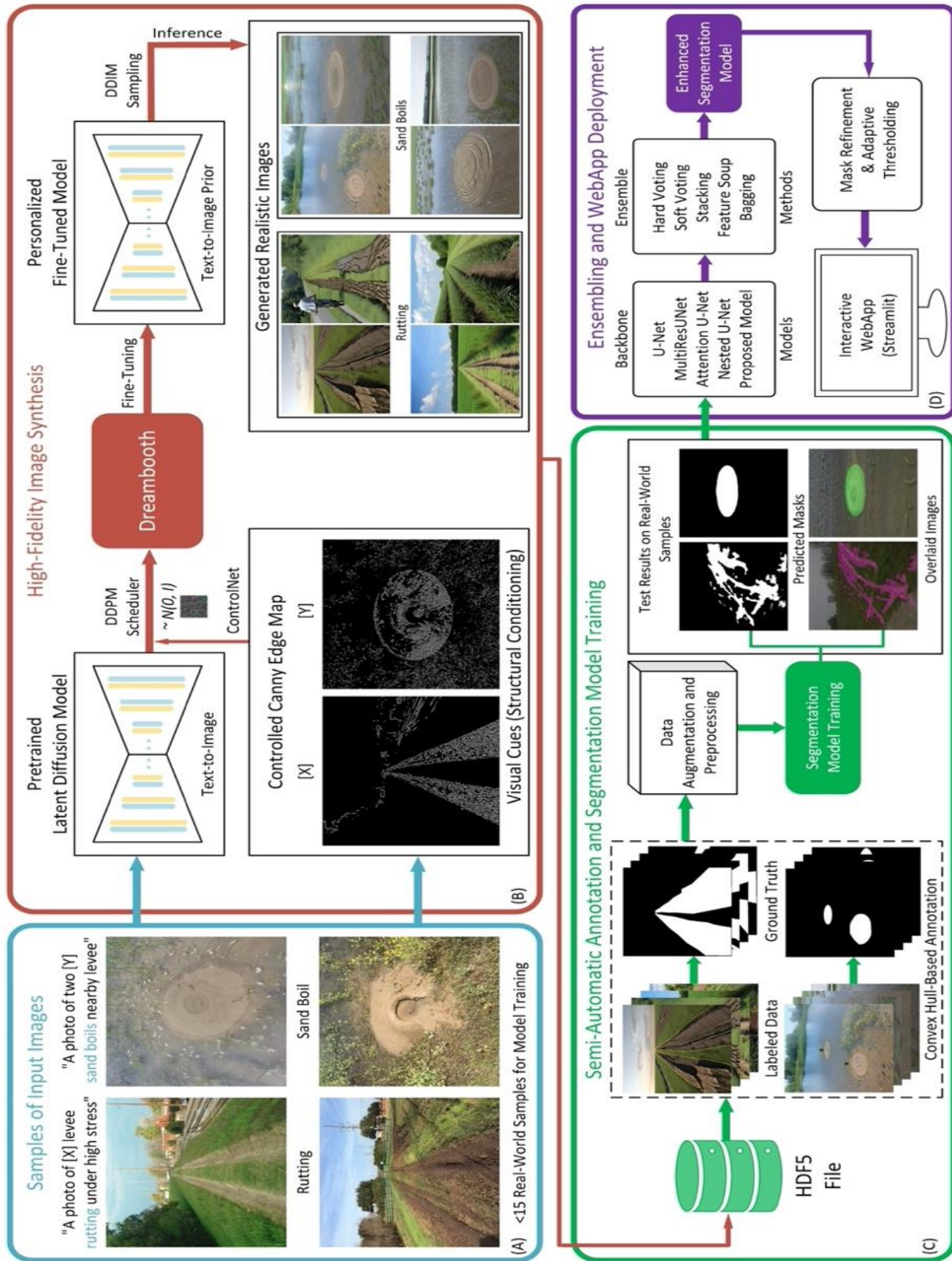
$$x_{t-1} = \frac{\sqrt{\alpha_{t-1}}}{\sqrt{\alpha_t}} \left( x_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t}} \cdot \epsilon_{\theta}(x_t, t) \right) + \sqrt{1 - \alpha_{t-1}} \cdot \epsilon_{\theta}(x_t, t) \quad (3.9)$$

Here,  $\alpha_t$  and  $\alpha_{t-1}$  control the noise schedule at respective timesteps, and  $\epsilon_{\theta}(x_t, t)$  is the predicted noise. The deterministic nature of DDIM thus enables efficient and rapid generation of coherent and high-quality synthetic defect images. Generation was further guided by carefully crafted textual prompts—including both instance-specific and broader class descriptors—and explicit negative prompts ("blurred," "low-resolution") to minimize undesirable artifacts. Prior preservation maintained a balance between defect-specific features and realism. The experimental setup for DreamBooth fine-tuning parameters is summarized in Table 1.

Table 3.1: Summary of DreamBooth Fine-Tuning Configuration and Parameters

Parameter	Setting
Base Model	Stable Diffusion v1.5 (Pretrained)
Instance Images	10 – 15 per defect type
Class Images	10 – 12x instance images
Resolution	512 × 512 pixels
Training Steps	~80 × instance images (800–3200)
Learning Rate	$1 \times 10^{-6}$
Optimizer	8-bit Adam
Sampling Method	DDIM (25–50 inference steps)
Precision	Mixed (FP16)
Hardware	NVIDIA A100 GPU (80GB)

This measured approach results in a fine-tuned diffusion model capable of synthesizing new images that remain coherent—often with natural variations in soil type, lighting, or environmental details—while accurately reproducing the distinctive characteristics observed in the limited training examples. For geological applications, this means creating diverse yet accurate representations of features like sand boils or rutting that can supplement limited field observations or enhance training datasets for automated detection systems. The flowchart for the high-fidelity image synthesis pipeline is presented in Figure 3.5, illustrating the overall DreamBooth fine-tuning process alongside segmentation model training and ensembling strategies, which are discussed in subsequent sections.

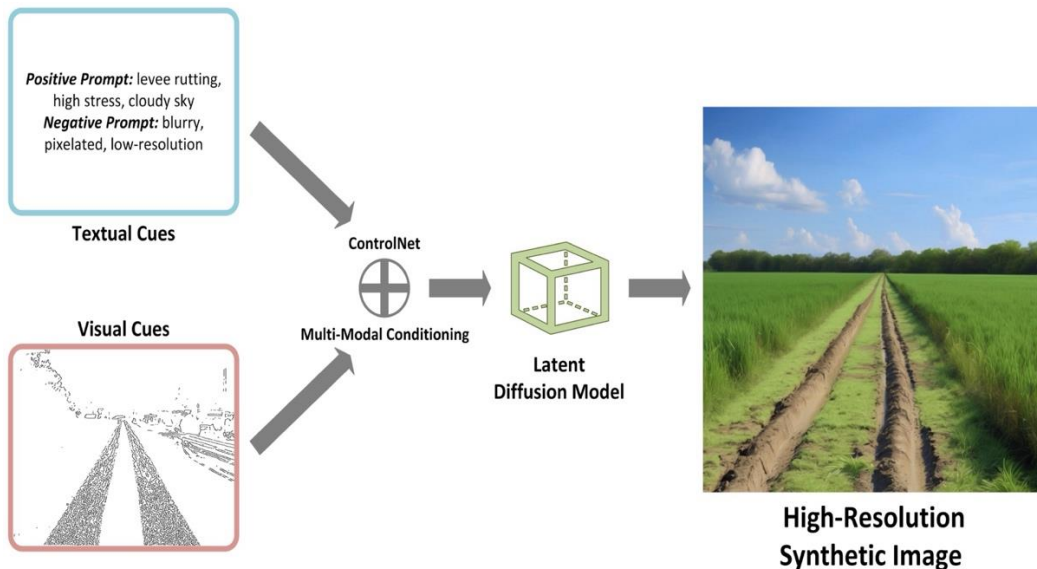


**Figure 3.5:** Overview of the end-to-end image synthesis and segmentation pipeline incorporating DreamBooth fine-tuning, structural conditioning via ControlNet, semi-automatic annotation, and model ensembling for enhanced levee defect detection.

### C. ControlNet for Structure-Guided Generation

Despite the advantages of DreamBooth in learning specialized features from limited images, text prompts alone can sometimes prove insufficient for controlling the spatial layout of generated scenes. This shortcoming becomes especially relevant in geotechnical contexts, where capturing the precise shape or structure of a defect—such as the curved contour of a sand boil or the elongated edges of a rut—is critical.

ControlNet addresses this gap by enabling structure-guided image generation, allowing users to provide additional visual cues, such as edge maps, along with refined textual inputs (positive and negative prompts). Positive prompts explicitly describe desired features (e.g., "levee rutting, high stress, cloudy sky"), guiding the model toward generating contextually accurate images, while negative prompts (e.g., "blurry, pixelated, low-resolution") discourage undesirable attributes, thereby improving image quality and realism. Through this combined textual-visual guidance, the model achieves stronger geometric fidelity and precise aesthetic control, translating into more accurate renditions of domain-specific forms. Unlike traditional diffusion models relying solely on text conditioning, ControlNet integrates visual structural information directly into the generative process, making it particularly valuable for engineering applications where geometric and visual accuracy are paramount.



**Figure 3.6:** Illustration of the ControlNet image generation pipeline combining textual (positive and negative prompts) and visual (edge map) guidance to produce realistic, structurally accurate images of levee rutting.

Figure 3.6 illustrates the integration of these textual and visual cues into the ControlNet generation pipeline. A user-provided textual prompt specifies the overall appearance and



conditions desired in the generated scene, while an auxiliary structural input, such as a Canny edge map derived from reference images, explicitly encodes critical geometric outlines. In the case of rutting defects, edge maps highlight the distinctive linear contours of depressions, while prompts help ensure accurate soil texture and environmental conditions. The model then jointly processes these cues, combining visual structure and textual descriptions within a pretrained diffusion framework to yield coherent, structurally faithful synthetic images.

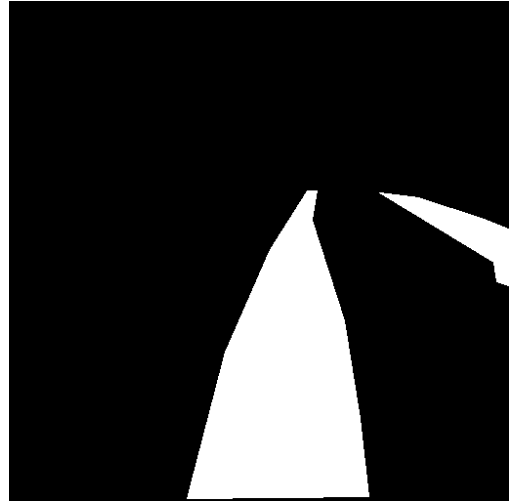
The underlying principle of ControlNet involves augmenting a standard diffusion process with these auxiliary inputs that represent desired structural constraints. By conditioning the diffusion model on explicit visual features and carefully curated text prompts, ControlNet effectively "locks in" the fundamental geometry and aesthetic attributes of the generated images. The architecture achieves this dual-level control by introducing trainable copy layers within the underlying U-Net backbone, allowing structural and textual information to seamlessly combine during generation.

Implementing ControlNet within the stable diffusion framework requires modest alterations to the standard pipeline. Users load both a pretrained base diffusion model and the specialized ControlNet model designed to handle structural guidance. At inference, carefully selected textual prompts describing desired and undesired image attributes are paired with structural inputs such as edge-detected images, typically produced using techniques like Canny edge detection. Hyperparameters including inference steps, guidance scales, and negative prompts offer further control over visual fidelity and detail.

For rutting and sand boil data augmentation specifically, this approach ensures preservation of distinctive defect shapes while allowing diverse visual variations, significantly reducing unrealistic artifacts. Practically, this method generates multiple realistic renditions of the same structural defect under varying environmental and soil conditions, greatly enhancing dataset diversity without compromising geometric fidelity. Maintaining shape accuracy is critical for segmentation tasks that depend heavily on precise pixel-level boundaries. ControlNet's capacity to integrate multiple modalities of guidance—both textual and structural—thus substantially strengthens the reliability and realism of generated defect imagery, ultimately bolstering the accuracy and performance of automated levee defect detection systems.

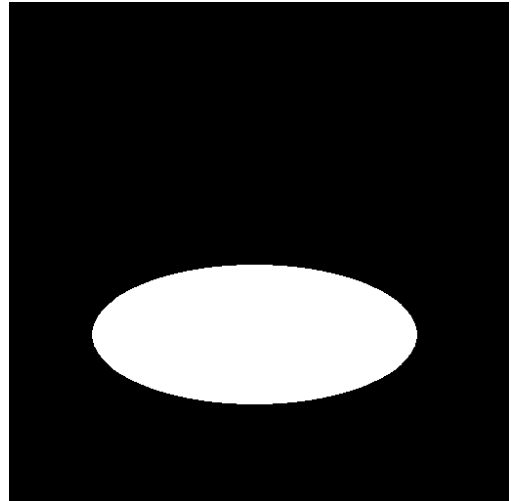
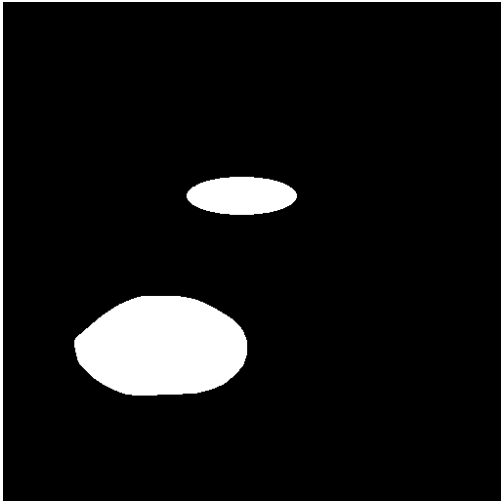
## 3.4 Data Annotation and Pre-Processing

Accurate pixel-level annotations are critical for semantic segmentation models, as they define the ground truth boundaries of rutting and sand boil features. Following best practices in computer vision research—akin to methods used for fault detection [47], we used the VGG Image Annotator (VIA) to mark rut boundaries and sand boils. Given the irregular, elongated nature of ruts, polygon tools were employed for careful boundary tracking, while ellipse tools were employed for sand boils due to its round appearance. Each marked rut was cross verified to ensure it accurately encompassed the depressed region without intruding into adjacent non-rutted surfaces. Figure 3.7 illustrates examples from the synthetically generated dataset, showcasing annotated rutting and sand boil defects across diverse backgrounds. Following annotation, we exported the data in JSON format which contained the precise coordinate information of all marked rut and sand boil regions. We then developed a custom Python script to process this JSON file, extracting the polygon and ellipse coordinate data to generate corresponding binary masks.



(a)

(b)



(c)

(d)

Figure 3.7: Annotated synthetic dataset samples, with (a), (b) showing rutting and (c), (d) depicting sand boils. Each set includes the original image, ground truth, and an overlaid segmentation mask, illustrating fault variations across different backgrounds.

Building upon these annotated masks, we segregated 20% of the images and their corresponding masks to form an independent test set, ensuring the model evaluation remains unbiased. The remaining corpus, however, was still limited for deep learning purposes, risking overfitting—a typical challenge when datasets are small. To address this, we adopted a comprehensive data augmentation strategy, drawing on proven techniques from similar fault-detection studies.

In total, 30 distinct augmentation methods were selected, spanning geometric (e.g., rotations, flipping, elastic transforms), spatial (e.g., optical and grid distortions), pixel-level (e.g., noise injection, superpixel and dropping pixel techniques), channel transformation such as Channel Shuffle and Contrast Limited Histogram Equalization (CLAHE), and filter-based manipulations (e.g., blur or sharpening). This multilayered approach expands both the dimensionality and variability of the training set, thereby improving the model’s capacity to generalize under diverse real-world conditions. Crucially, transformations that altered the spatial layout—such as flips or elastic deformations—were also applied to the masks to preserve alignment, whereas purely color-related transformations were restricted to the images alone. Certain techniques (e.g., random cropping, padding, and simulated snow) were excluded upon closer inspection, as they proved detrimental to highlighting the subtle boundaries of rutting and sand boil defects.

After finalizing the augmentation pipeline, the augmented data was further partitioned into 70% for training and 30% for validation, maintaining a common seed for reproducibility. Additionally, each image was resized to  $512 \times 512$  and scaled to the  $[0,1]$  range by dividing pixel intensities by 255. These preprocessing steps not only ensure consistency and compatibility with standard deep learning architectures but also bolster the model’s overall performance. By systematically combining high-fidelity annotations, targeted data augmentation, and careful dataset partitioning, we significantly increased both the volume and diversity of the training samples—ultimately laying a robust foundation for accurate segmentation of rutting and sand boil regions.

To further enhance the model’s generalization ability and minimize false positives, the training process incorporated a dedicated phase using negative images—scenes that contain no visible levee defects. These included clean terrains such as grassy levee tops, undisturbed embankments, and water channels without anomalies. After initial training on annotated defect regions, the model was subsequently trained using these negative examples, enabling it to better differentiate between faulty and non-faulty areas. This two-stage training approach allowed the model to build a stronger contrastive understanding, significantly improving its ability to reject false detections during inference and ensuring greater robustness in real-world deployment scenarios.

### 3.5 Automated Convex Hull-Based Annotation

Accurate annotation of defect images, especially those generated through sophisticated generative models such as DreamBooth and ControlNet, remains essential yet often labor-intensive and arduous task for semantic segmentation workflows. To streamline and automate this

crucial process, an Automated Convex Hull-Based Annotation pipeline was developed. This method leverages computational geometry—specifically, the convex hull algorithm—to systematically and precisely generate segmentation masks, significantly accelerating dataset preparation and reducing manual labeling efforts while preserving annotation consistency and precision.



Figure 3.8: Illustration of the Convex Hull Algorithm applied to a finite set of points. The convex hull (highlighted in red) represents the smallest convex polygon encompassing all given points. In image segmentation, this concept is leveraged to automatically generate precise annotations by identifying and encapsulating the spatial extent of defect regions.

At its core, the convex hull algorithm mathematically computes the smallest convex polygon encompassing a finite set of points in two-dimensional space. The practical utility of this approach in image segmentation is evident, as it enables automated annotation by encapsulating the spatial extent of detected defects within a minimal enclosing boundary, as depicted in Figure 3.8. Formally, given a set of points  $P = \{p_1, p_2, \dots, p_n\}$ , the convex hull  $H$  is defined as the set of all convex combinations of these points here in Equation 3.10:

$$H = \{ \sum_{i=1}^n \alpha_i p_i \mid (\forall i: \alpha_i \geq 0), \sum_{i=1}^n \alpha_i = 1 \} \quad (3.10)$$

This means any point inside the convex hull is a weighted average of the original points, where all weights  $\alpha_i$  are non-negative and sum to 1. Geometrically, this ensures that the constructed polygon tightly wraps around the outermost points without concavities or internal holes. In the context of image segmentation, especially for defect detection using models like SandBoilNet, each predicted defect region in a binary mask corresponds to a discrete set of pixel coordinates. By treating these pixels as the point set  $P$ , the convex hull algorithm computes a minimal enclosing polygon that represents the outer boundary of each detected anomaly.

This formulation ensures that the annotated mask encloses the full spatial extent of the defect while eliminating noisy, fragmented edges. Moreover, since the convex hull includes all convex combinations of the detected pixel locations, the resulting polygon is mathematically guaranteed to be both tight and complete—capturing the anomaly boundary with minimal overreach. This not only improves annotation precision but also enhances consistency and efficiency for downstream tasks like training semantic segmentation networks.



Figure 3.9: Manual adjustment of convex hull annotations using the VGG Image Annotator (VIA). From left to right: (a) the original synthetic image showing a sand boil defect, (b) convex hull-based auto-annotation visualized as a red polygon, and (c) human refinement by dragging the polygon points for precise defect boundary alignment.

Practically, the annotation pipeline initiates by leveraging pretrained semantic segmentation models to generate preliminary masks for synthetic images. These masks undergo adaptive thresholding to produce binary representations of the segmented anomalies. Connected component labeling then identifies discrete defect regions, which are individually assessed based on area (thresholded at 100 pixels) to filter out insignificant artifacts. For each validated region, the convex hull is computed using efficient image processing routines (`convex_hull_image`) provided by the ‘`skimage`’ library [57]. Contours are subsequently extracted via OpenCV’s contour detection (`cv2.findContours`) function, delivering pixel-level polygon boundaries precisely outlining the defects.

Once polygon coordinates are established, annotations are visually validated by overlaying distinctively colored convex hull boundaries onto the original synthetic RGB images. Furthermore, the polygon coordinates are serialized into structured JSON files compatible with standard annotation tools (e.g., VGG Image Annotator). These generated JSON annotations facilitate rapid, manual quality checks and minor adjustments within interactive annotation tools, ensuring flexibility and accuracy validation without re-annotating images from scratch. This semi-automated workflow significantly reduces annotation time while retaining human oversight for critical corrections. As shown in Figure 3.9, various examples of sand boil anomalies are automatically enclosed using the convex hull polygon, which users can manually edit by dragging control points to better align with the ground truth.

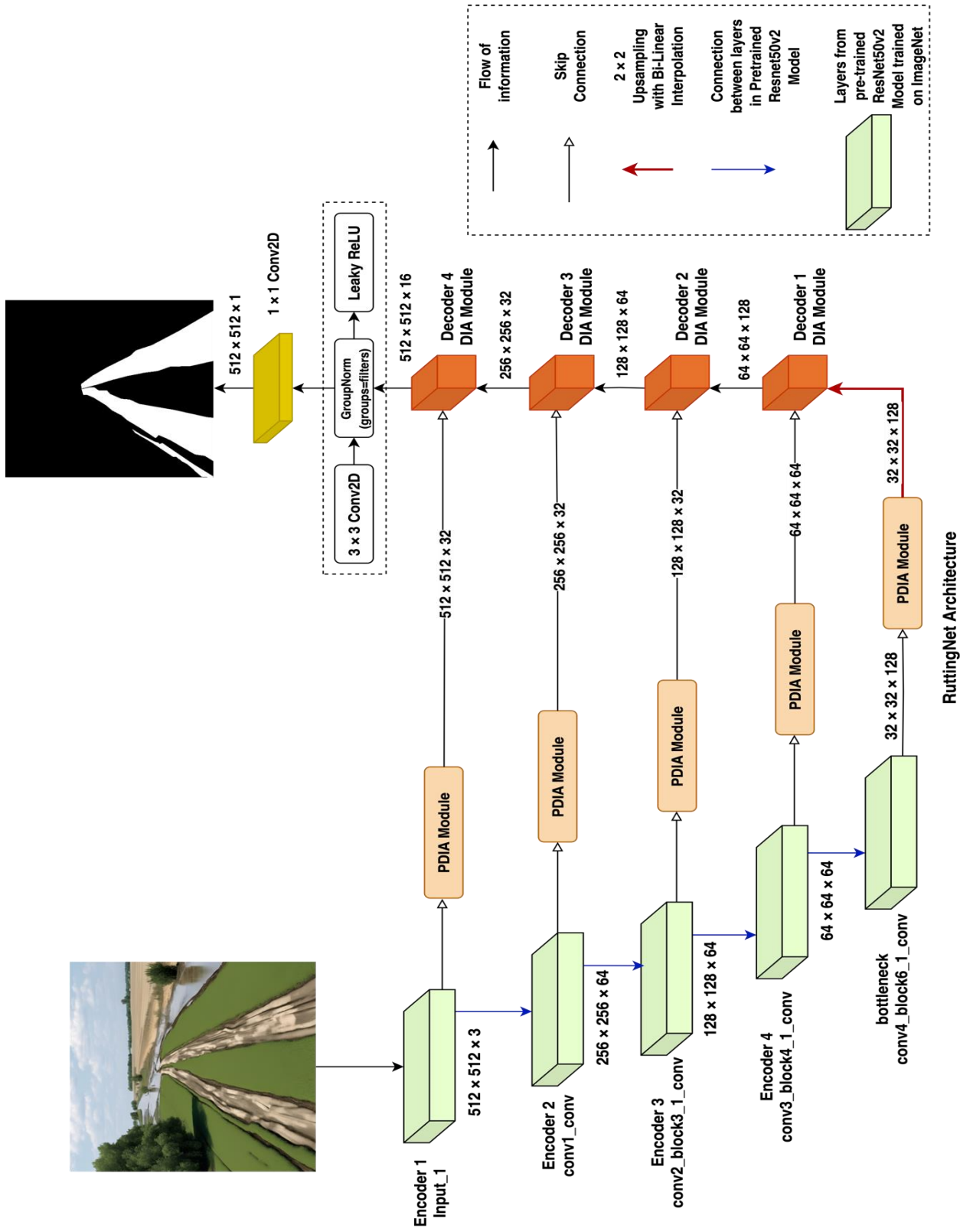
## 3.6 Baseline and Existing Models

This section outlines the baseline and existing segmentation models explored in this study, with a primary focus on partial fine-tuning using the ResNet50V2 architecture. Given the limited availability of annotated training data for levee defect segmentation—especially for rare faults like rutting and sand boils—this research adopts a transfer learning strategy that leverages the representational power of a pretrained ResNet50V2 backbone, originally trained on the large-scale ImageNet dataset. This approach enables controlled feature extraction and selectively generates feature maps most relevant to the fault segmentation domain, thereby enhancing performance.

The proposed baseline architecture utilizes a partial fine-tuning approach, wherein the early layers of the pretrained ResNet50V2 model are frozen and used purely for feature extraction, while the last 48 layers, including the bottleneck block, are fine-tuned on the domain-specific dataset. While originally built for classification tasks and not having a decoder on its own, this strategy selectively updates specific layers of a pre-trained ResNet50v2 architecture, retaining generalizable low and mid-level visual features learned from extensive datasets such as ImageNet. The rationale behind this design stems from the role of initial layers as fixed feature extractors in learning general-purpose low-level features (e.g., edges, textures, and corners), which are largely transferable across domains preserving valuable generic information. In contrast, the deeper layers at and beyond the bottleneck encode more task-specific semantic features and are thus adapted to the nuanced visual characteristics during fine-tuning to specialize in levee fault detection.

To preserve the spatial information necessary for pixel-level segmentation, the architecture adopts an encoder-decoder format, similar to the U-Net framework. Establishing a baseline model is crucial, as it provides a standardized benchmark for objectively evaluating improvements offered by more advanced architectures. Direct skip connections are employed from early encoder blocks to corresponding decoder layers, facilitating the transfer of fine-grained spatial details. This architectural design enables accurate delineation of subtle levee anomalies, such as thin rutting lines or irregular sand boils, particularly in challenging environmental textures. A schematic overview of this encoder-decoder architecture with integrated ResNet50V2 layers is shown in Figure 3.10 and Figure 3.11. The baseline model incorporates skip connections that merge detailed low-level encoder features with corresponding decoder stages, thereby preserving the fine spatial information essential for accurate segmentation.

The implementation of targeted partial fine-tuning also involves careful consideration of optimization stability, effectively mitigating overfitting risks posed by limited training datasets and ensuring robust adaptation to the levee domain. A low learning rate is used to prevent abrupt changes in pretrained representations, and batch normalization layers are frozen to preserve internal normalization statistics from the source domain (ImageNet). This is essential because the domain shift from ImageNet to geotechnical imagery can lead to inconsistent normalization behavior if these layers are retrained. Consequently, freezing batch normalization layers during fine-tuning helps retain generalization capacity while avoiding overfitting, especially when training on small datasets.

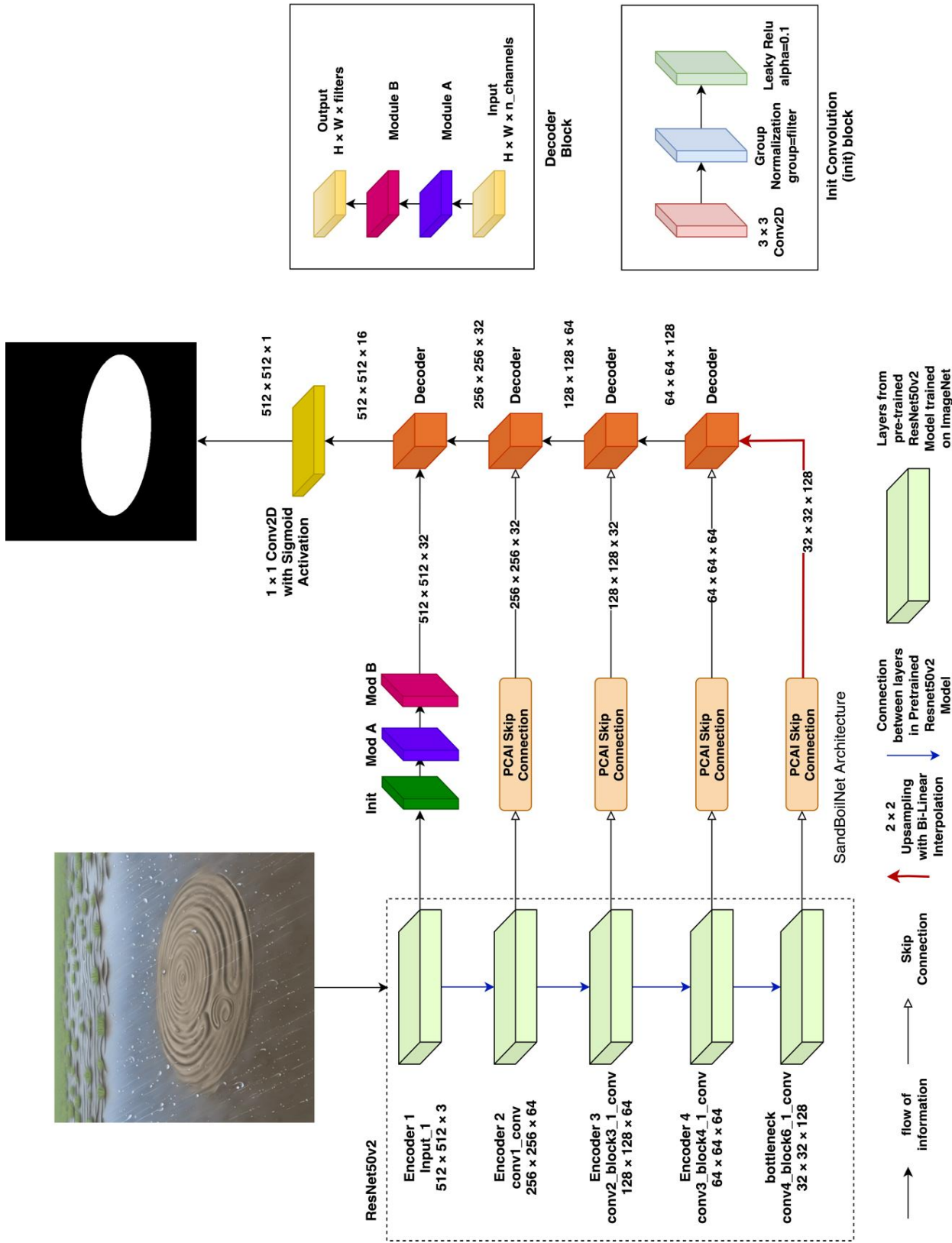


**Figure 3.10:** Proposed RuttingNet architecture featuring an encoder-decoder structure with skip connections. The model incorporates a PCA-Depthwise Inception Attention (PDIA) module within each skip connection, enabling multi-scale feature fusion and enhanced spatial attention for precise levee rutting segmentation.



Establishing this baseline is methodologically important, as it offers a consistent benchmark against which the performance of more advanced models can be both quantitatively and qualitatively compared. To build upon this foundation, several state-of-the-art segmentation models were explored with the objective of enhancing generalization performance in levee defect detection. Among them, MultiResUNet was considered for its ability to handle scale variance through multi-resolution residual blocks. Attention U-Net was incorporated for its integration of attention gates, which help emphasize spatially relevant features during decoding. U-Net++, known for its densely connected skip pathways, was also evaluated for its capacity to enhance feature reusability and improve segmentation precision across complex textures.

These architectures were selected based on their demonstrated success in related tasks, such as crack detection and seepage mapping in civil infrastructure monitoring. Furthermore, design choices and hyperparameters were informed by previous work on sand boil segmentation, ensuring domain relevance. Collectively, by combining partial fine-tuning strategies with carefully chosen model architectures and grounding improvements in a reliable baseline, this research constructs a progressive performance pipeline tailored to the challenges of automated levee fault monitoring.



**Figure 3.11:** Proposed SandBoilNet architecture featuring an encoder-decoder structure with PCAI-enhanced skip connections. The model leverages multi-scale feature fusion and attention-guided refinement, trained on a hybrid dataset combining real and synthetic sand boil imagery for improved segmentation performance.

The SandBoilNet architecture employed in this study is designed to handle the complexity and variability inherent in the task of sand boil segmentation. Built upon a resilient encoder-decoder framework, the model is augmented with PCAI (Principal Component Analysis-based Channel-Spatial Attention with Inception) Skip Connection Blocks, strategically positioned to reinforce the feature propagation between corresponding encoder and decoder layers. These blocks serve a dual purpose: first, by applying PCA-based dimensionality reduction, they eliminate redundant or noisy feature dimensions, leading to a more compact and discriminative representation; second, by embedding channel-spatial attention mechanisms, they enable the model to emphasize critical spatial locations and feature channels associated with sand boil characteristics. This guided attention is particularly beneficial for enhancing the network's sensitivity to small, irregularly shaped, or low-contrast sand boils, which might otherwise be suppressed or diluted in conventional skip connection pathways.

The model's internal inception-inspired modules further improve feature representation by capturing multi-scale context through parallel convolutional operations. These modules enable the network to learn both fine and coarse details simultaneously, facilitating more accurate segmentation across diverse scenarios—whether the sand boils are partially occluded, embedded in noisy textures, or appear in varying shapes and scales. The decoder progressively reconstructs spatial resolution by fusing semantically rich, attention-weighted features from deeper layers with refined details preserved in the skip pathways.

A key distinction in this work lies in the training strategy, which utilizes a hybrid dataset composed of both real sand boil imagery and a large collection of synthetically generated samples. The synthetic data was created to mirror real-world variation in background, lighting, texture, and environmental context, ensuring that the model is exposed to a wide range of visual patterns. This synthetic-real fusion is further enhanced through a comprehensive set of augmentation techniques (detailed previously), which simulate real-world distortions and scene variability. This approach not only addresses the limitations posed by the relatively small number of annotated real samples but also ensures that the model generalizes effectively across different field conditions.

By combining multi-scale representation, PCA-augmented attention, and a diverse, hybrid training dataset, the SandBoilNet model achieves strong segmentation performance in visually complex levee environments. Its architecture and data-driven training methodology together contribute to improved fault localization accuracy, offering a scalable and reliable solution for automated levee health monitoring.

### 3.7 Metrics and Loss Functions

The accurate segmentation and localization of levee faults are critical for effective infrastructure monitoring and preventive maintenance. The performance of deep learning-based semantic segmentation models, particularly in specialized applications such as rut and sand boil detection, heavily relies on the careful selection of appropriate evaluation metrics and loss functions. Due to the subtle nature and irregular geometry of rutting defects, relying solely on basic pixel accuracy is insufficient and can misrepresent a model's true segmentation capability, especially in imbalanced datasets where defect pixels constitute a very small portion compared to background pixels.

Therefore, multiple metrics were employed to comprehensively assess the models' effectiveness in precisely identifying and delineating rutting regions. Among these, the

Intersection over Union (IoU) and the Dice Coefficient (DC) stand out as particularly suitable, as both provide insights into the spatial congruence between predicted and ground truth segmentation masks. The IoU, defined in Equation 3.11, measures the overlap between predicted and ground truth pixels, effectively quantifying how accurately the model captures the precise boundaries of rutting. Meanwhile, the Dice Coefficient, defined in Equation 3.12, offers a complementary evaluation, calculating the harmonic mean of precision and recall, and providing balanced insight into the model's accuracy and sensitivity toward defect segmentation.

$$\text{IoU} = \frac{Y_{\text{predicted}} \cap Y_{\text{gt}}}{Y_{\text{predicted}} \cup Y_{\text{gt}}} \quad (3.11)$$

$$\text{Dice Coefficient (DC)} = \frac{2 \cdot |Y_{\text{predicted}} \cap Y_{\text{gt}}|}{|Y_{\text{predicted}}| + |Y_{\text{gt}}|} \quad (3.12)$$

However, given the significant class imbalance in levee rutting and sand boils data—characterized by a considerably higher proportion of background pixels relative to defect pixels—standard evaluation metrics alone might not fully capture the model's performance. To address this challenge, Balanced Accuracy (BA) was also utilized, as it equally considers sensitivity (true positive rate) and specificity (true negative rate), as defined in Equations 3.13 – 3.15. Balanced accuracy provides a more representative measure of the segmentation model's ability to correctly classify both defect and background regions.

$$\text{Balanced Accuracy (BA)} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \quad (3.13)$$

$$\text{Sensitivity or True Positive Rate (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3.14)$$

$$\text{Specificity or True Negative Rate (TNR)} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (3.15)$$

$$\text{Macro F1 Score (MaF1)} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.16)$$

$$\text{BCE Loss} = - (Y_{\text{gt}} \cdot \log(Y_{\text{p}}) + (1 - Y_{\text{gt}}) \cdot \log(1 - Y_{\text{p}})) \quad (3.17)$$

$$\text{Dice Loss} = (1 - \text{DC}) \quad (3.18)$$

$$\text{BCE Dice Loss} = \theta_1 \cdot \text{BCE Loss} + \theta_2 \cdot \text{Dice Loss} \quad (3.19)$$

Therefore, composite loss functions, such as the BCE-Dice Loss, were explored to enhance training efficacy. This combined loss function integrates Binary Cross Entropy (BCE) Loss and Dice Loss, as defined in Equations 3.17 and 3.18, thereby simultaneously encouraging accurate pixel-level classifications and promoting overlap similarity. The weighted combination of these two loss components, as expressed in Equation 3.19, effectively balances the optimization process by addressing both precise boundary delineation and overall region segmentation. In addition to loss-based evaluation, Micro and Macro F1 scores were used to assess segmentation performance from both localized and overall perspectives. Micro F1 calculates precision and recall per instance—similar to the Dice coefficient on an image-wise basis—before averaging across the dataset. In contrast, the Macro F1 Score, defined in Equation 3.16, provides a class-agnostic average of precision and recall over the entire test set, offering a more holistic measure of model performance in imbalanced scenarios.

In the above equations,  $Y_{\text{predicted}}$  and  $Y_{\text{gt}}$  denote the predicted and ground-truth pixel sets, while TP, FP, TN, and FN represent the respective counts of true-positive, false-positive, true-negative, and false-negative pixels. The variable  $Y_{\text{p}}$  corresponds to the predicted probability for

the rutting and sand boil class. In Equation 3.19,  $\theta_1 = 0.5$  and  $\theta_2 = 0.5$  are the weights for BCE and Dice losses. Collectively, the strategic use of these metrics and loss functions ensures robust evaluation and effective optimization of segmentation models. By addressing the distinctive challenges of levee defects segmentation—particularly class imbalance and subtle defect visibility—this comprehensive evaluation framework enhances model performance and reliability, ultimately providing meaningful insights for improved levee maintenance strategies.

## 3.8 Experimental Setup

The segmentation models for levee rutting and sand boils were implemented using the Keras deep learning framework [53], a user-friendly and high-level neural network API, integrated within the TensorFlow ecosystem. To leverage modern computing capabilities, model training was conducted on four of NVIDIA’s A100 80GB GPUs, taking advantage of their high-speed parallel processing and optimized tensor operations. To ensure efficient and consistent training across multiple devices, a distributed training approach was employed using TensorFlow’s MirroredStrategy. This strategy enables synchronous training by replicating the model across multiple GPUs, where gradients are averaged, and variables are updated in lockstep, maintaining uniform learning across all replicas. By utilizing MirroredStrategy, training efficiency was significantly improved, and larger batch sizes were accommodated without sacrificing model performance.

To maintain consistency in weight initialization and ensure fair comparisons across different architectures, all convolutional layers were initialized using the He initialization method. He initialization is designed explicitly for layers utilizing rectified linear unit (ReLU) activations, ensuring optimal variance preservation across network layers and facilitating faster convergence during training. By initializing the network weights from a carefully scaled random distribution, the He initialization mitigates problems such as vanishing or exploding gradients, which are common in deep neural networks. Furthermore, applying a fixed random seed ensured uniformity across the initial network states and guaranteed identical compositions of training and validation datasets across experimental runs. This approach minimized variability stemming from random weight assignments and dataset partitioning, thus significantly enhancing the reproducibility and comparability of experimental results.

All segmentation models were trained for a maximum of 200 epochs, employing a batch size of 32 and an initial learning rate of  $4e-4$ . To further reduce the risk of overfitting and promote stable model training, an early stopping criterion was integrated, terminating training when the validation loss showed no improvement over eight consecutive epochs. Additionally, a learning rate scheduler was activated when validation loss plateaued, applying a decay factor of 0.06 every six epochs to refine model optimization and encourage convergence progressively.

Throughout training, both computational efficiency and inference performance were closely monitored. Training and inference durations were systematically recorded, enabling an objective comparison of computational requirements across different model configurations. Ultimately, the best-performing model was identified and selected based on achieving the highest Dice Coefficient (DC) on the validation set, prioritizing accurate delineation of rutting and sand boil defects. This checkpointed model was retained and subsequently evaluated on an independent test set, ensuring that reported results reliably reflected the model's generalization capabilities in realistic, unseen scenarios.

# Chapter 4: Ensemble Learning and Deployment

## 4.1 Ensemble Learning for Enhanced Segmentation

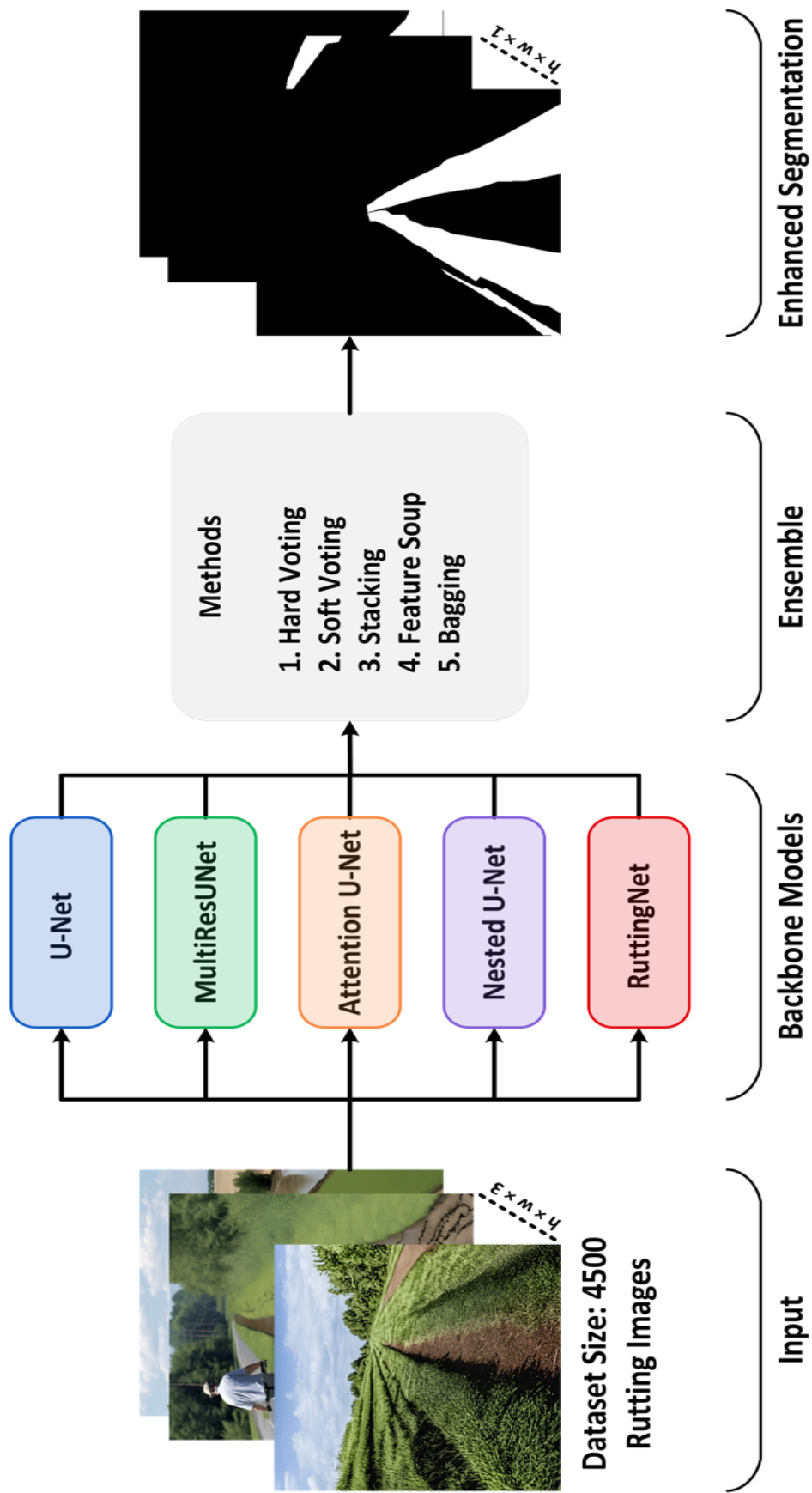
Segmentation accuracy in levee fault detection can significantly benefit from combining predictions of multiple deep learning models—a practice known as ensemble learning. Individual models often exhibit unique strengths and limitations due to differences in architecture, training dynamics, and learned feature representations. By strategically aggregating the outputs of several segmentation models, ensemble methods aim to capitalize on the collective predictive power of these diverse architectures, mitigating overfitting, individual weaknesses, and increasing overall prediction reliability and accuracy.

In this research, multiple prominent U-Net-based architectures—including the standard U-Net, MultiResUNet, Attention U-Net, Nested U-Net (U-Net++), and our specifically designed RuttingNet—were employed as backbone models for the ensemble. Each of these architectures independently generates segmentation masks, capturing varying levels of spatial detail and emphasizing different aspects of rutting features.

To integrate the outputs of multiple segmentation models, various ensemble strategies were explored. Majority voting assigns the final class label to each pixel based on the most frequent prediction among models, effectively reducing outlier effects and enhancing segmentation reliability. Weighted averaging, on the other hand, refines this process by assigning different importance levels to each model's prediction, with higher weights given to models demonstrating superior performance on validation data. This ensures that the most reliable models contribute more significantly to the final segmentation. Median stacking further enhances robustness by selecting the median probability value for each pixel, mitigating the influence of extreme outlier predictions. Additionally, geometric mean stacking incorporates a multiplicative fusion of predicted probabilities, reinforcing agreement among models and downplaying inconsistent outputs.

Beyond direct prediction aggregation, this study also employs feature soup, a technique that blends intermediate feature representations from different models before passing them through a shared decoder. This method allows the model to harness diverse learned feature representations, enriching the segmentation process by capturing more nuanced spatial and textural details. Furthermore, bagging (bootstrap aggregating) was utilized to train multiple models on different randomly sampled subsets of the training data, ensuring better generalization by exposing models to varied distributions. By systematically evaluating these ensemble techniques—ranging from simpler voting methods to advanced stacking and feature-level blending strategies—this study comprehensively explored methods for maximizing segmentation accuracy and robustness against variations in lighting, surface textures, and environmental conditions.

The ensemble strategy implemented in this research is illustrated conceptually in Figure 4.1, highlighting the input flow through backbone models, the employed ensemble aggregation techniques, and the final enhanced segmentation output. By clearly understanding and harnessing these ensemble techniques, the presented methodology significantly advances the effectiveness of automated levee monitoring systems, ensuring reliable, accurate, and practically robust rutting segmentation performance.



**Figure 4.1:** Conceptual Diagram of the Ensemble Learning Approach Integrating Multiple U-Net Variants and Aggregation Strategies for Enhanced Rutting and SandBoil Segmentation.

Employing ensemble methods allowed the aggregation of diverse strengths across multiple segmentation architectures, resulting in predictions that were consistently superior to any single model’s outputs. The final selected ensemble strategy represents an optimal balance of computational efficiency, accuracy, and practical applicability for real-world levee rutting inspection tasks.

In addition to the ensemble strategies outlined above, weighted average stacking with thresholding emerged as the most effective method in this study for post-ensemble mask refinement. After computing a weighted average of the probability maps generated by each model—based on their individual performance on validation data—a fixed threshold of 0.5 was applied to convert the soft ensemble outputs into binary segmentation masks. This two-step approach not only ensured that higher-confidence predictions were more influential in the final mask but also enabled fine-tuning of the segmentation boundary sharpness through threshold optimization. The use of thresholding following weighted averaging allowed for a calibrated control over sensitivity and specificity, reducing false positives while preserving critical rutting features. Empirical evaluation demonstrated that this method consistently outperformed other fusion strategies in terms of Intersection over Union (IoU) and Dice coefficient, especially under varying lighting and surface texture conditions. As such, weighted averaging with thresholding was adopted as the preferred ensemble method due to its superior performance, interpretability, and ease of implementation within real-time levee inspection pipelines.

## 4.2 WebApp Deployment for Defect Segmentation

Effective deployment of AI-driven segmentation models is crucial for translating research outcomes into practical tools for levee inspection. A real-time web-based application was developed using the Streamlit framework as shown in Figure 4.2, which offers an interactive and accessible user interface for field inspectors and decision-makers. This deployment integrates multiple advanced functionalities, including model inference, image preprocessing, thresholding, overlap resolution, and visual post-processing, ensuring efficient visualization and analysis of levee defects like sand boils, seepage, rutting, cracks, potholes, and vegetation encroachment.

To achieve robust and responsive performance, TensorFlow and Keras models were integrated into the Streamlit app, which can be inferred on any personal computer without needing any heavy GPUs for acceleration. The system adopts dynamic GPU memory management by enabling memory growth, which helps efficiently allocate and free GPU resources during inference—crucial for maintaining real-time responsiveness when processing large video streams or batches of high-resolution images.

To further optimize user experience and app responsiveness, the web application adopts lazy loading and caching mechanisms. Lazy loading ensures that heavy operations—such as model loading and preprocessing—are only executed when explicitly required. This approach significantly reduces initial page load time, particularly when multiple models are available for selection. Additionally, Streamlit’s “@st.cache\_resource” and “@st.cache\_data” decorators are used to store model weights and preprocessed image results, preventing redundant re-computations and enabling near-instantaneous response for previously seen inputs. Together, these optimizations enhance the app’s scalability and performance, especially during iterative testing or high-frequency inspections in the field.



The application supports both single-image analysis and video-based processing. Upon uploading an image or video file, the app preprocesses the input according to model-specific requirements. This preprocessing pipeline involves resizing images to a consistent input dimension 512×512 pixels, followed by customizable preprocessing such as brightness and contrast adjustments, Gaussian blurring, and optional edge detection using the Canny algorithm. Additional augmentations such as image rotation, horizontal or vertical flipping, and resolution scaling can be interactively controlled via the Streamlit interface, providing flexibility to field inspectors who often deal with varied environmental conditions and image quality.

Inference within the web application utilizes multiple segmentation models, each trained specifically for identifying particular defects. Users select the defect types through a dynamic sidebar interface. Predictions generated by the models produce initial probability masks, which are then refined through a thresholding mechanism to create binary masks clearly delineating defect boundaries. Three thresholding methods are available: a manual threshold (user-defined), Otsu's automatic thresholding, and percentile-based adaptive thresholding. Otsu's method provides automated thresholding optimized by minimizing intra-class variance, whereas percentile-based methods dynamically adjust thresholds according to statistical distribution of prediction scores, ideal for varied defect types and uncertain lighting conditions.

A crucial aspect of real-world segmentation tasks is handling overlapping or adjacent detections. This app employs several post-processing strategies to resolve such conflicts with precision:

- **Non-Maximum Suppression (NMS):** Bounding box-based visualization was further refined using the Non-Maximum Suppression (NMS) algorithm [58], which is designed to reduce overlapping predictions by retaining only the most confident bounding box in a cluster of nearby detections. Without NMS, as shown in Figure 4.3, multiple overlapping boxes appear around small rutting defects, leading to visual clutter and ambiguity. In contrast, Figure 4.4 illustrates how NMS effectively suppresses redundant detections in seepage segmentation, resulting in cleaner and more interpretable bounding box outputs. This enhancement improves visual clarity and supports efficient, field-friendly inspection workflows.
- **Confidence Thresholding and Distance Control:** Binary masks generated from probabilistic outputs are filtered using confidence scores to retain only the most reliable and accurate detections. To manage cases where different defect regions appear close to one another, morphological dilation is applied to establish proximity zones around each defect. If two masks fall within a configurable distance threshold, the system identifies them as potentially overlapping, enabling automatic suppression of redundant or conflicting segmentations. This mechanism reduces misclassification and ensures clear separation between nearby fault regions, improving the precision of the overall segmentation output, as visually demonstrated in Figure 4.6.

These combined techniques ensure clear, non-redundant delineation of complex coexisting defects, reducing false positives, significantly improving interpretability, and practical utility in levee assessments.

Visual post-processing further improves the interpretability of model predictions. The app supports two primary visualization methods: transparent overlays and bounding-box annotations. Transparent overlays apply distinct colors to each defect mask with adjustable transparency to

clearly visualize the defect locations and boundaries. Bounding boxes simplify defect interpretation, particularly for quick visual scans or when precise pixel-level details are less critical. Each defect type is color-coded—green for sand boils, pink for seepage, blue for cracks, and so forth—enhancing quick defect recognition.

For video inputs, real-time processing was implemented to support continuous frame-by-frame analysis with adjustable playback speed, ensuring efficient monitoring. Users can define the starting point within the video, with the application dynamically managing inference timing to sustain smooth and coherent real-time visualization. The processed frames are stored temporarily, compiled into a downloadable video file upon completion, enabling subsequent review or detailed inspections by engineers.

To complement these capabilities, we also integrated a lightweight active-learning feedback loop directly into the web app. After the model produces its initial defect masks, users can enter a “Human-in-the-Loop Re-annotation” mode in which they correct or refine mask boundaries via an interactive Streamlit canvas. All user edits are captured as timestamped JSON entries (defect type + polygon coordinates), producing a curated set of expert-labeled failure cases. These annotations can then be ingested into an offline retraining pipeline—scheduled nightly or weekly—to iteratively improve the ensemble segmentation models without burdening the real-time UI with on-the-fly training. This design shown in Figure 4.5 ensures continuous model refinement informed by actual field corrections, striking a balance between responsiveness in deployment and quality gains from active-learning.

The entire web-based framework was developed to be interactive, intuitive, and accessible without advanced technical knowledge, bridging the gap between AI-driven analytics and practical levee inspection. This comprehensive system significantly accelerates defect identification, enhances precision through visual and algorithmic refinement, and provides an intuitive, scalable solution suitable for field deployment in diverse environments.

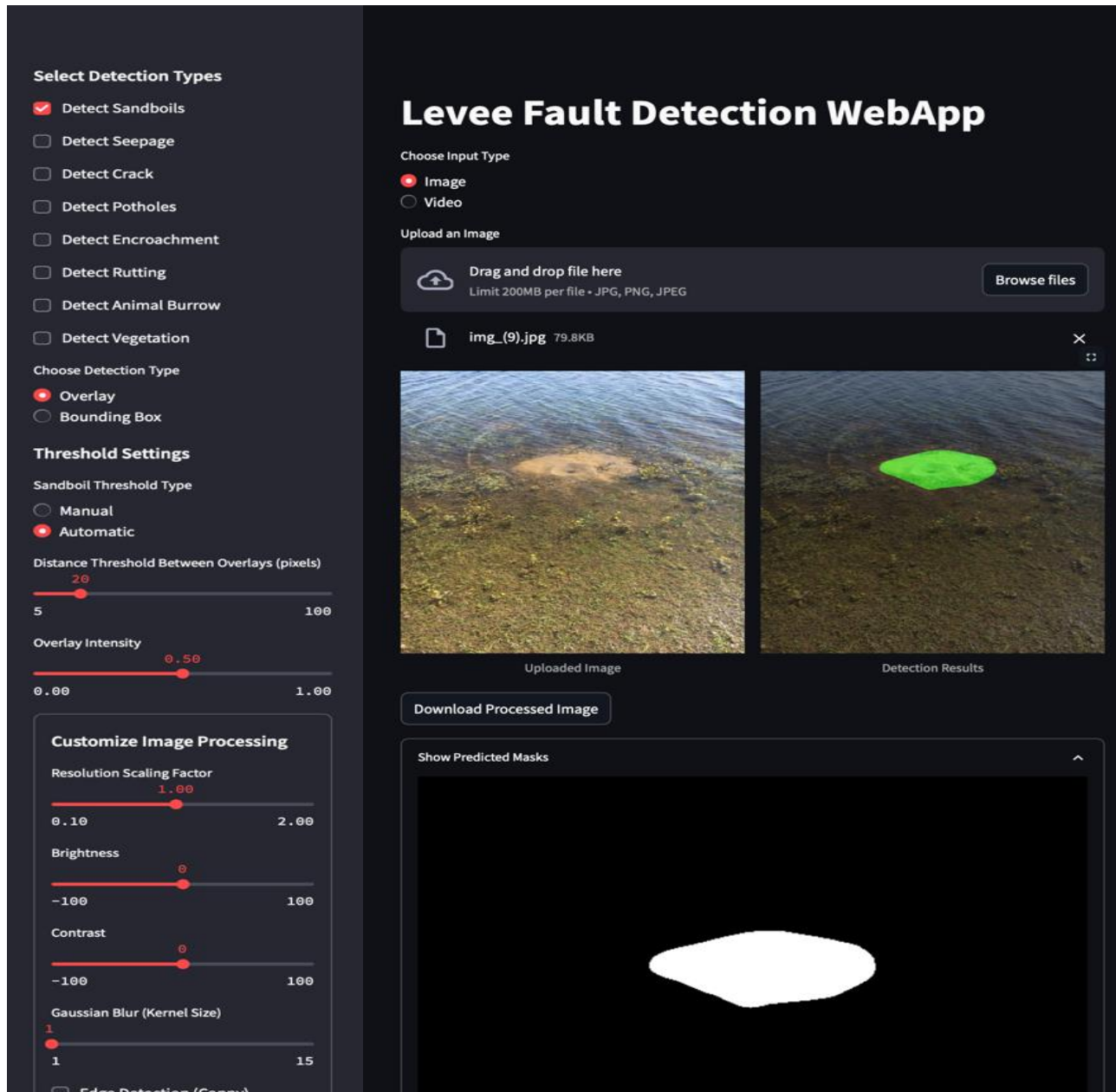


Figure 4.2: Sand Boil Detection Interface in the Levee Fault Detection WebApp with Overlay and Mask Visualization. The sidebar enables users to select defect types, choose between overlay or bounding box visualization, and adjust thresholding and preprocessing parameters such as brightness, contrast, blur, and resolution scaling—allowing flexible, real-time customization to enhance segmentation accuracy under diverse field conditions.

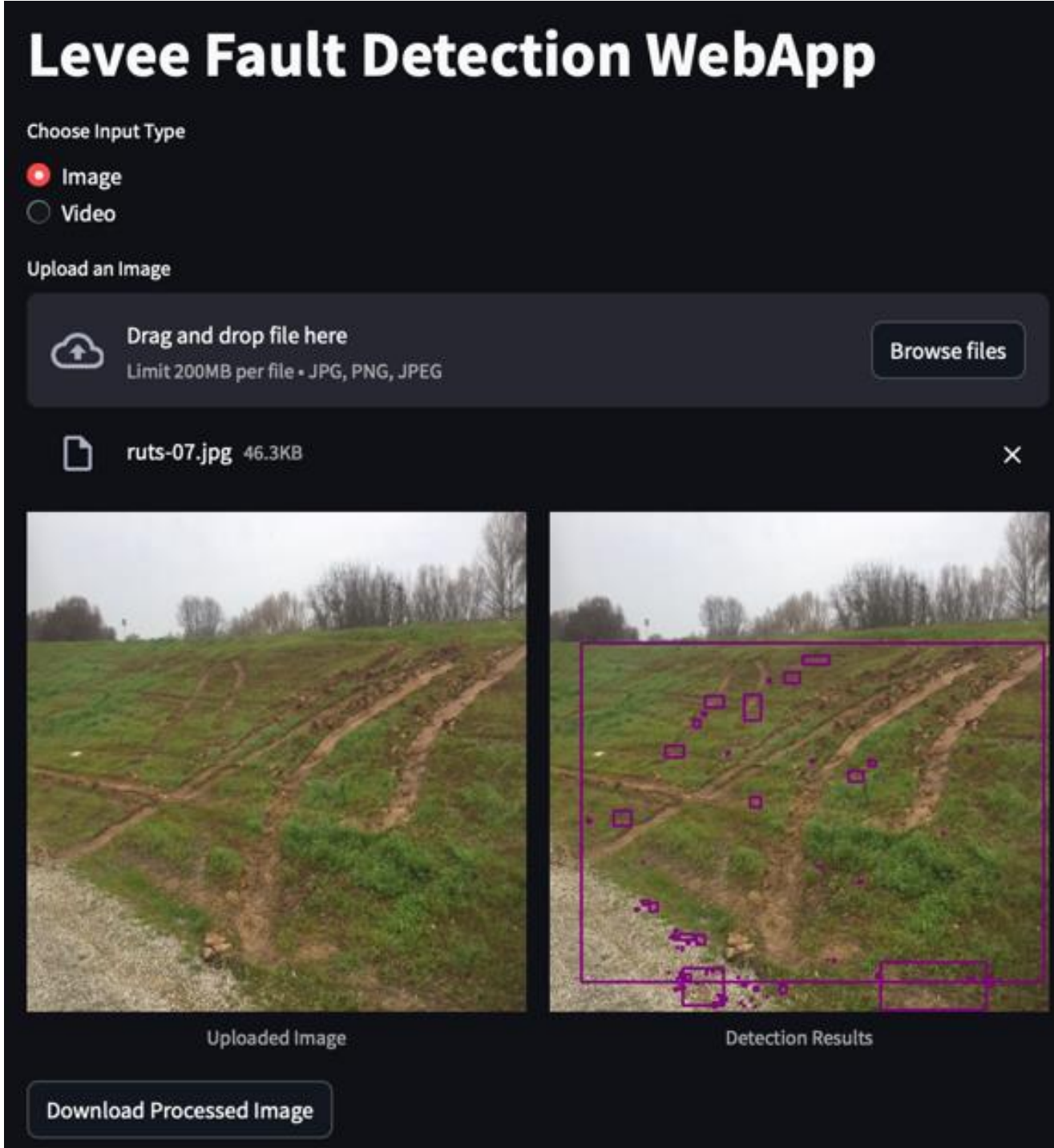


Figure 4.3: Rutting Detection Interface in the Levee Fault Detection WebApp Displaying Bounding Box Annotations for Surface Fault Localization without non-max suppression algorithm.

# Levee Fault Detection WebApp

Choose Input Type

Image

Video

Upload an Image



Drag and drop file here

Limit 200MB per file • JPG, PNG, JPEG

Browse files



UPNT18Mar16-001\_20180328100637.jpg 210.7KB




Uploaded Image



Detection Results


Download Processed Image

Figure 4.4: Seepage Detection Interface in the Levee Fault Detection WebApp with Bounding Box Annotations Highlighting Moisture-Induced Fault Zones with non-max suppression algorithm.

 Do you want to re-annotate this overlay?

## Human-in-the-Loop Re-annotation

Which detection are you correcting?

Sandboil 



Delete Last Annotation



Clear All Annotations

**Instructions:** Click to place points. **Double-click** to close the polygon. Use buttons above to delete.



Save Annotations



Saved 2 annotation(s)!

Figure 4.5: Human-in-the-Loop Re-annotation interface in the deployed WebApp, showing sandboil and seepage overlays corrected by expert input and saved as JSON for active-learning feedback.



Figure 4.6: Segmentation overlay visualization from the deployed web application showing simultaneous detection of sand boil (green) and seepage (pink) defects with clearly separated masks and no overlapping, demonstrating effective multi-defect segmentation and priority-based overlap resolution.

## Chapter 5: Results and Analysis

This chapter presents a comprehensive evaluation of the proposed deep learning framework for rutting and sand boil segmentation in levee systems. Both quantitative and qualitative assessments are conducted to provide a holistic view of model performance, including comparisons to established baselines, insights into computational costs, and a discussion of strengths and limitations.

Models were assessed using a dedicated test set of levee images containing annotated rutting or sand boil defects. Key performance indicators included Balanced Accuracy (BA), Intersection over Union (IoU), Dice Coefficient (DC), Precision, Recall, and Macro F1. These metrics collectively address common challenges—such as severe class imbalance—by capturing how well the models detect small, irregularly shaped defects without neglecting the extensive background region. Table 5.1 and Table 5.2 summarize the performance results from standard baselines (U-Net, Attention U-Net, U-Net++, MultiResUNet) and the specialized models developed specifically for rutting (RuttingNet) and sand boils (SandBoilNet). Notably, the proposed specialized architectures outperformed classic baselines, demonstrating improvements of approximately 8–13 percentage points in IoU and 6–10% in Balanced Accuracy, highlighting their robust capability in correctly classifying both defective and non-defective pixels.

These performance improvements are significantly influenced by transfer learning and generative augmentation techniques. Models leveraging pretrained backbones and enriched synthetic data from DreamBooth or ControlNet consistently demonstrated faster convergence and higher accuracy compared to training from scratch. This underscores the effectiveness of targeted fine-tuning and synthetic data augmentation in compensating for scarce real-world datasets typical of specialized fields like levee inspection.

All models were trained on four NVIDIA A100 GPU using a distributed training approach. By employing selective fine-tuning alongside mixed-precision (FP16) training, advanced architectures completed training within approximately 30–40 minutes for 200 epochs—even when incorporating computationally demanding multi-scale and attention-based modules. During inference, the specialized models achieved an encouraging processing speed averaging around 0.7–1.0 seconds per  $512 \times 512$  image, making the approach practical for real-time scenarios, including drone-based inspections, and continuous field-based camera monitoring.

Figure 5.1 and Figure 5.2 provide key qualitative comparisons among the baseline and specialized models across diverse real-world scenarios. Specialized models such as RuttingNet and SandBoilNet demonstrated enhanced proficiency in capturing subtle defect edges and fine textural details, effectively addressing the common issue of fragmented or incomplete predictions produced by baseline U-Net variants in visually complex contexts. Although occasional false positives persisted—especially in reflective or muddy conditions—multi-scale feature extraction and attention mechanisms significantly reduced their occurrence, improving the reliability of segmentation predictions. Moreover, the specialized architectures effectively managed scenarios involving multiple coexisting anomalies by clearly distinguishing boundaries between different defect types, despite occasional minor confusions in areas with extremely subtle transitions.



Practically, achieving Intersection over Union values above 50% and Balanced Accuracy scores surpassing 80% significantly validates the proposed framework's suitability for large-scale levee monitoring. Among all models, the ensemble model—particularly the weighted average stacking approach with threshold tuning—yielded the highest performance gains, further boosting IoU and Balanced Accuracy by approximately 3–5 percentage points. This ensemble strategy effectively leveraged the complementary strengths of individual models, resulting in more stable and accurate segmentation outcomes. The near real-time processing capability enhances its utility, allowing inspectors to swiftly identify areas needing closer examination. The robust nature of segmentation results, supported by synthetic data augmentation, suggests strong generalizability across varied environmental conditions, including fluctuating lighting, diverse soil textures, and different moisture states.

Overall, these comprehensive results affirm that the integrated approach—encompassing advanced neural architectures, targeted transfer learning, generative augmentation, and semi-automated annotation—markedly surpasses traditional CNN-based baselines. While certain challenges remain, particularly in handling reflective surfaces and overlapping defects, the achieved gains in accuracy, speed, and general reliability represent meaningful progress toward automated levee inspection systems. Looking ahead, future improvements such as lightweight model optimizations using quantization, pruning and distillation for edge-device deployment, ensemble learning strategies, semi-supervised augmentation methods, and incorporating explainable AI techniques could further enhance the system’s robustness, scalability, and field applicability. Collectively, these developments promise substantial advancements in proactive maintenance strategies, ultimately contributing to improved flood protection and strengthened infrastructure resilience.

Table 5.1: Metric results of models on Rutting Levee test dataset. The best metrics results are shown in bold. The model with the highest Intersection over Union (IoU) score is indicated in bold and underlined.

Models	IoU Score	Dice Coefficient	Mean IoU	Binary Accuracy	Balanced Accuracy
U-Net	0.450	0.565	0.474	0.651	0.669
MultiResUnet	0.400	0.534	0.435	0.603	0.665
Attention U-Net	0.430	0.561	0.442	0.609	0.669
U-Net++	0.430	0.561	0.442	0.609	0.669
Proposed RuttingNet	0.549	0.690	0.600	0.749	0.786
Weighted Ensemble	0.582	0.749	0.631	0.770	0.812

Table 5.2: Metric results of models on the SandBoil Levee test dataset. The best metric results are shown in bold. The model with the highest Intersection over Union (IoU) score is indicated in bold and underlined.

Models	IoU Score	Dice Coefficient	Mean IoU	Binary Accuracy	Balanced Accuracy
U-Net	0.385	0.503	0.651	0.922	0.744
MultiResUnet	0.447	0.570	0.684	0.927	0.786
Attention U-Net	0.410	0.536	0.662	0.924	0.766
U-Net++	0.430	0.561	0.674	0.925	0.775
Proposed SandBoilNet	0.496	0.645	0.625	0.805	0.839
Weighted Ensemble	0.547	0.671	0.697	0.871	0.850

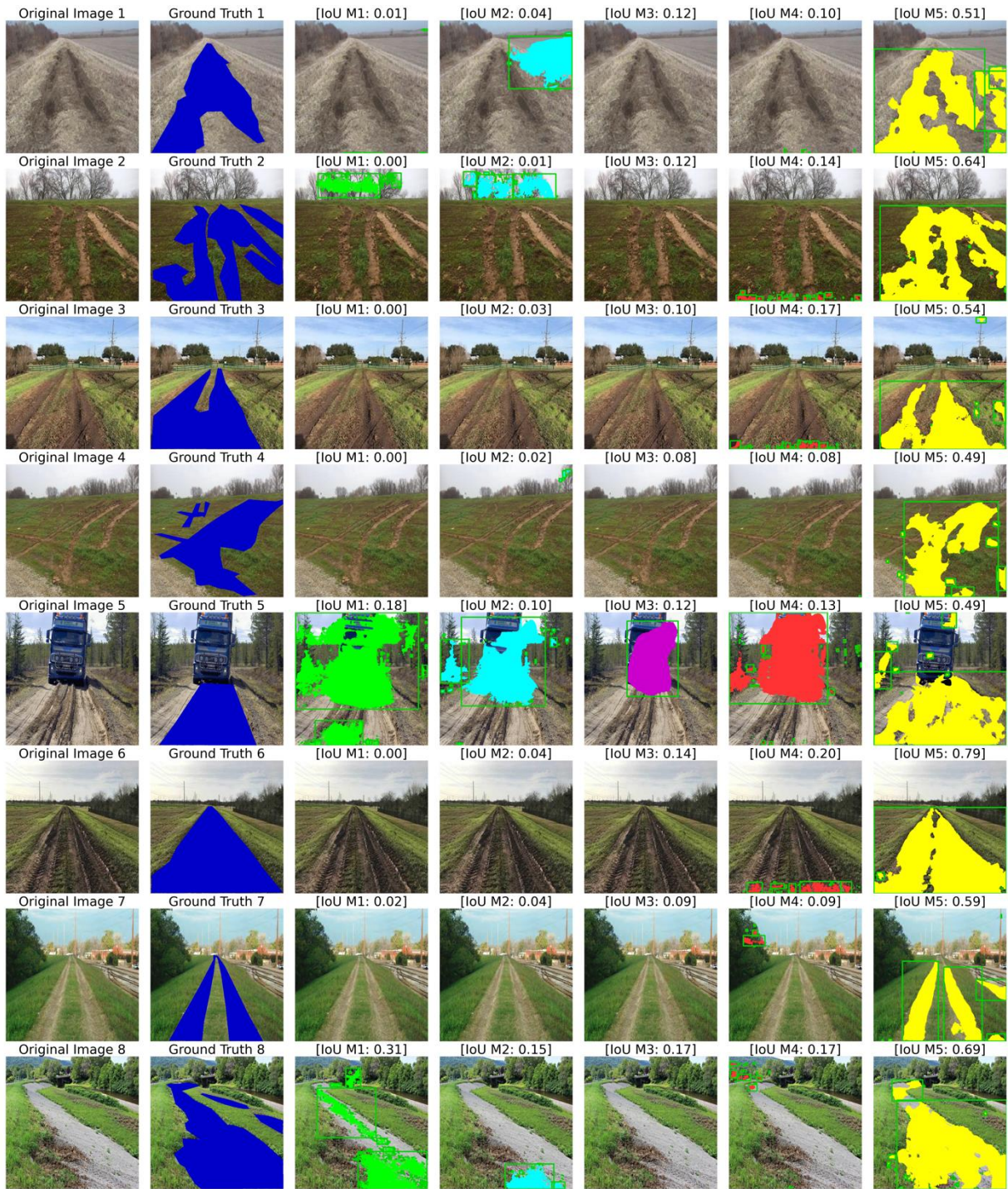


Figure 5.1: Segmentation results on levee images illustrating rutting defects. The blue segmentation mask represents the ground truth annotations overlaid on the original images. Predictions from U-Net (green), MultiResUNet (cyan), Attention U-Net (purple), U-Net++ (red), RuttingNet dataset (yellow), and the final ensemble model utilizing Weighted Average Stacking with thresholding (white) are displayed sequentially.

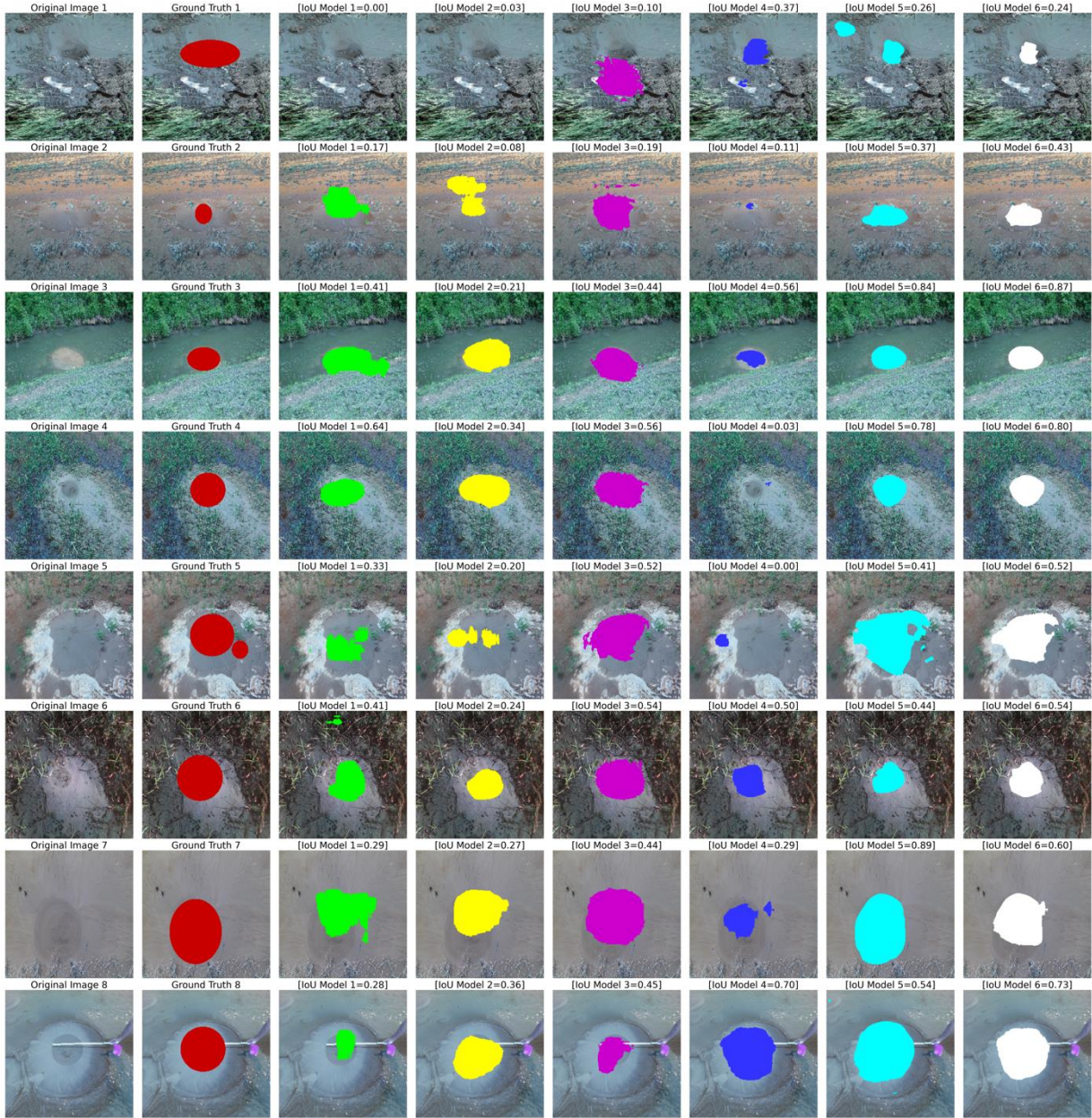


Figure 5.2: Segmentation results on levee images illustrating sand boil defects. The red segmentation mask represents the ground truth annotations overlaid on the original images. Predictions from U-Net (green), MultiResUNet (yellow), Attention U-Net (purple), U-Net++ (dark blue), SandBoilNet trained on a hybrid synthetic dataset (light blue), and the final ensemble model utilizing Weighted Average Stacking with thresholding (white) are displayed sequentially.

# Chapter 6: Concluding Remarks

## 6.1 Conclusion

This research presents an end-to-end deep learning framework for automated levee defect detection, integrating advanced segmentation architectures, generative augmentation techniques, and a real-time deployment pipeline to address the longstanding challenge of limited annotated data in geotechnical domains. Focused on structurally complex and rare anomalies—such as rutting, seepage, and sand boils—the study leverages a multi-tiered strategy encompassing StyleGAN2-ADA for early GAN-based experiments, DreamBooth and ControlNet for diffusion-driven generation, and custom semantic segmentation architectures (RuttingNet, SandBoilNet) designed for high-resolution, fine-grained defect delineation.

By harnessing the capabilities of DreamBooth and ControlNet under the Stable Diffusion framework, the research successfully generated diverse, high-fidelity training samples from minimal input images (10–15 per class). These synthetic samples preserved key structural and textural properties of the target defects, significantly enriching the available dataset, improving generalization, and mitigating overfitting—a critical need given the scarcity of real-world levee imagery. The generative pipeline allowed for scalable dataset augmentation across various environmental contexts, contributing to improved model robustness and segmentation accuracy.

This research contributes not only to the niche field of levee monitoring but also offers generalizable insights into the combined use of generative augmentation, diffusion models, and semantic segmentation in other domains where data scarcity and structural variability remain significant challenges. By focusing on realism-preserving augmentation, modular network design, and efficient deployment pipelines, the study establishes a scalable blueprint for future AI-driven environmental monitoring systems.

## 6.2 Future Work

Despite the promising outcomes achieved, several directions remain open for future enhancement and exploration. One crucial aspect is model optimization for edge deployment. While current models perform well on high-end GPUs, deploying them on lightweight or embedded devices such as UAVs, mobile phones, or IoT-based monitoring systems requires significant size and speed optimization. Techniques like model pruning—which eliminates redundant weights or neurons—can help compress the model without major sacrifices in accuracy. Similarly, quantization-aware training (QAT) and post-training quantization (PTQ) can convert models from 32-bit floating-point to 8-bit integers, drastically reducing memory footprint and computational demand, thereby enabling inference on resource-constrained hardware.

Another promising avenue is the use of stacked generalization (model stacking) or ensemble learning to improve robustness and generalization across varied environmental conditions and image resolutions. By combining the strengths of multiple models—such as a base segmentation model with auxiliary attention-based refiners or geometric-aware decoders (meta

model)—stacking strategies can reduce bias and variance, thereby improving segmentation consistency. Integrating lightweight transformer modules or deformable convolutions within existing encoder-decoder pipelines can also improve attention to localized features, particularly useful in identifying small, irregular anomalies like animal burrows or shallow rutting.

Moreover, expanding the dataset through semi-supervised or self-supervised learning methods remains a priority. Techniques such as pseudo-labeling or contrastive learning could help exploit unlabeled data or weak labels to further enhance model performance. Finally, incorporating explainable AI (XAI) mechanisms would aid in interpreting model outputs, fostering user trust and supporting field engineers in decision-making processes. These future advancements, when combined, can drive the development of a fully autonomous, scalable, and interpretable levee defect monitoring system—applicable not only in civil infrastructure but in broader geospatial and remote sensing applications.

The proposed approach shows strong segmentation performance but struggles with distinguishing visually similar defects, especially in complex environments with lighting variations, occlusions, or surface textures. While it uses semantic segmentation, it lacks instance differentiation. Integrating panoptic or instance segmentation could improve defect identification and localization, but this would require additional labeled datasets, increasing the annotation burden. These challenges point to the need for more advanced segmentation techniques for better real-world defect characterization.

# References

- [1] U.S. Army Corps of Engineers. (n.d.). Levee owner's manual for non-federal flood control works. U.S. Army Corps of Engineers, Rock Island District. Retrieved February 17, 2025, from <https://www.mvr.usace.army.mil/Portals/48/docs/EC/LSP/LeveeOwnersManual.pdf>.
- [2] W. M. Leavitt and J. J. Kiefer, "Infrastructure interdependency and the creation of a normal disaster: The case of hurricane katrina and the city of new orleans," *Public works management & policy*, vol. 10, no. 4, pp. 306–314, 2006.
- [3] USACE, A summary of risks and benefits associated with the usace levee portfolio, 2018.
- [4] U.S. Army Corps of Engineers. Periodic Inspection Report: Dallas Floodway, Trinity River, Dallas, Dallas County, Texas. Report No. 9, 3–5 Dec. 2007, CiteSeerX, <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=da0976af55c985d83ea7ebd7b1dd47f904b49b49>.
- [5] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, Jul. 2022, doi: 10.1109/TPAMI.2021.3059968.
- [6] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, Springer, 2015, pp. 234–241.
- [7] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82 031–82 057, 2021.
- [8] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
- [9] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.
- [10] F. Zhuang et al., "A Comprehensive Survey on Transfer Learning," in *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021, doi: 10.1109/JPROC.2020.3004555.
- [11] N. Ruiz et al., "Dreambooth: Fine-Tuning Text-to-Image Diffusion Models for Subject-Driven Generation," *CVPR*, 2023.
- [12] L. Zhang et al., "Adding Conditional Control to Text-to-Image Diffusion Models," *arXiv:2302.05543*, 2023.
- [13] N. Ibtehaz and M. S. Rahman, "Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation," *Neural networks*, vol. 121, pp. 74–87, 2020.
- [14] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis*

- and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4, Springer, 2018, pp. 3–11.
- [15] O. Oktay, J. Schlemper, L. L. Folgoc, et al., “Attention u-net: Learning where to look for the pancreas,” arXiv preprint arXiv:1804.03999, 2018.
- [16] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, “A review of semantic segmentation using deep neural networks,” *Int J Multimed Info Retr*, vol. 7, no. 2, pp. 87–93, Jun. 2018, doi: 10.1007/s13735-017-0141-z.
- [17] S. Hao, Y. Zhou, and Y. Guo, “A Brief Survey on Semantic Segmentation with Deep Learning,” *Neurocomputing*, vol. 406, pp. 302–321, Sep. 2020, doi: 10.1016/j.neucom.2019.11.118.
- [18] D. Feng et al., “Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges,” in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, March 2021.
- [19] A. Khanal, R. Rizk and K. Santosh, “Ensemble Deep Convolutional Neural Network to Identify Fractured Limbs using CT Scans,” 2023 IEEE Conference on Artificial Intelligence (CAI), Santa Clara, CA, USA, 2023, pp. 156–157, doi: 10.1109/CAI54212.2023.00075.
- [20] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, “Deep learning in remote sensing applications: A meta-analysis and review,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 152, pp. 166–177, Jun. 2019, doi: 10.1016/j.isprsjprs.2019.04.015.
- [21] J. Long, E. Shelhamer, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440. Accessed: Feb. 15, 2025. [Online]. Available: [https://openaccess.thecvf.com/content\\_cvpr\\_2015/html/Long\\_Fully\\_Convolutional\\_Networks\\_2015\\_CVPR\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2015/html/Long_Fully_Convolutional_Networks_2015_CVPR_paper.html)
- [22] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid Scene Parsing Network,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2881–2890. Accessed: Feb. 15, 2024. [Online]. Available: [https://openaccess.thecvf.com/content\\_cvpr\\_2017/html/Zhao\\_Pyramid\\_Scene\\_Parsing\\_CVPR\\_2017\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2017/html/Zhao_Pyramid_Scene_Parsing_CVPR_2017_paper.html)
- [23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778. Accessed: Feb. 16, 2024. [Online]. Available: [https://openaccess.thecvf.com/content\\_cvpr\\_2016/html/He\\_Deep\\_Residual\\_Learning\\_CVPR\\_2016\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html)
- [24] 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, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.
- [25] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully



- Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2699184.
- [26] M. Panta, M. T. Hoque, M. Abdelguerfi and M. C. Flanagin, "IterLUNet: Deep Learning Architecture for Pixel-Wise Crack Detection in Levee Systems," in *IEEE Access*, vol. 11, pp. 12249-12262, 2023, doi: 10.1109/ACCESS.2023.3241877.
- [27] Kuchi, Aditi, Md Tamjidul Hoque, Mahdi Abdelguerfi, and Maik C. Flanagin. "Machine learning applications in detecting sand boils from images." *Array* 3 (2019): 100012.
- [28] Panta, Manisha, Md Tamjidul Hoque, Kendall N. Niles, Joe Tom, Mahdi Abdelguerfi, and Maik Falanagin. "Deep Learning Approach for Accurate Segmentation of Sand Boils in Levee Systems." *IEEE Access* 11 (2023): 126263-126282.
- [29] Panta, Manisha, Padam Jung Thapa, Md Tamjidul Hoque, Kendall N. Niles, Steve Sloan, Maik Flanagin, Ken Pathak, and Mahdi Abdelguerfi. "Application of Deep Learning for Segmenting Seepages in Levee Systems." *Remote Sensing* 16, no. 13 (2024): 2441.
- [30] Alshawi, Rasha, Md Meftahul Ferdous, Mahdi Abdelguerfi, Kendall Niles, Ken Pathak, and Steve Sloan. "Imbalance-Aware Culvert-Sewer Defect Segmentation Using an Enhanced Feature Pyramid Network." *arXiv preprint arXiv:2408.10181* (2024).
- [31] Alshawi, Rasha, Md Tamjidul Hoque, and Maik C. Flanagin. "A depth-wise separable U-Net architecture with multiscale filters to detect sinkholes." *Remote Sensing* 15, no. 5 (2023): 1384.
- [32] Kingma, Diederik P. "Auto-encoding variational bayes." *arXiv preprint arXiv:1312.6114* (2013).
- [33] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial networks." *Communications of the ACM* 63, no. 11 (2020): 139-144.
- [34] Mirza, Mehdi. "Conditional generative adversarial nets." *arXiv preprint arXiv:1411.1784* (2014).
- [35] Dai, Yimian, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. "Attentional feature fusion." In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 3560-3569. 2021.
- [36] Tran, Ngoc-Trung, Viet-Hung Tran, Bao-Ngoc Nguyen, Linxiao Yang, and Ngai-Man Man Cheung. "Self-supervised gan: Analysis and improvement with multi-class minimax game." *Advances in neural information processing systems* 32 (2019).
- [37] "GAN 2.0: NVIDIA's Hyperrealistic Face Generator". *SynchedReview.com*. December 14, 2018. Retrieved October 3, 2019.
- [38] Karras, Tero, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. "Analyzing and improving the image quality of stylegan." In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110-8119. 2020.
- [39] Dhariwal, Prafulla, and Alexander Nichol. "Diffusion models beat gans on image synthesis." *Advances in neural information processing systems* 34 (2021): 8780-8794.

- [40] Zhang, Dejin, Xin Xu, Hong Lin, Rong Gui, Min Cao, and Li He. "Automatic road-marking detection and measurement from laser-scanning 3D profile data." *Automation in Construction* 108 (2019): 102957.
- [41] Salmivaara, Aura, Mikko Miettinen, Leena Finér, Samuli Launiainen, Heikki Korpunen, Sakari Tuominen, Jukka Heikkonen et al. "Wheel rut measurements by forest machine-mounted LiDAR sensors—accuracy and potential for operational applications?." *International Journal of Forest Engineering* 29, no. 1 (2018): 41-52.
- [42] Cao, Minh-Tu, Kuan-Tsung Chang, Ngoc-Mai Nguyen, Van-Duc Tran, Xuan-Linh Tran, and Nhat-Duc Hoang. "Image processing-based automatic detection of asphalt pavement rutting using a novel metaheuristic optimized machine learning approach." *Soft Computing* 25, no. 20 (2021): 12839-12855.
- [43] Arezoumand, Sara, Ahmadreza Mahmoudzadeh, Amir Golroo, and Barat Mojaradi. "Automatic pavement rutting measurement by fusing a high speed-shot camera and a linear laser." *Construction and Building Materials* 283 (2021): 122668.
- [44] Faisal, Ali, and Suliman Gargoum. "Automated Assessment of Pavement Rutting Using Mobile LiDAR Data." In presentation at the Innovations in Pavement Management, Engineering and Technologies Session, TAC Conference & Exhibition, Ottawa, ON. 2023.
- [45] Maser, Ken, and Adam Carmichael. *Ground penetrating radar evaluation of new pavement density*. No. WA-RD 839.1. Washington (State). Dept. of Transportation. Office of Research and Library Services, 2015.
- [46] U.S. Army Corps of Engineers. 2022. "Flood Damage Reduction Segments/Systems Inspection Report Bennington, Roaring Branch Left Bank Levee Inspection Summary." North Atlantic Division. Accessed February 23, 2025. <https://cms5.revize.com/revize/bennington/Document%20Center/Government/Documents%20and%20Reports/ACE%20Flood%20Damage%20Red%20Report%202022.pdf>.
- [47] A. Dutta and A. Zisserman, "The via annotation software for images, audio and video," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 2276–2279.
- [48] Padam Jung Thapa, Md Tamjidul Hoque, September 14, 2024, "Synthetic Sand Boil Dataset for Levee Monitoring", IEEE Dataport, doi: <https://dx.doi.org/10.21227/x8m8-5693>.
- [49] Aanstoos, James V., Khaled Hasan, Charles G. O'Hara, Saurabh Prasad, Lalitha Dabbiru, Majid Mahrooghy, Balakrishna Gokaraju, and Rodrigo Nobrega. "Earthen levee monitoring with synthetic aperture radar." In *2011 IEEE applied imagery pattern recognition workshop (AIPR)*, pp. 1-6. IEEE, 2011.
- [50] Guan, Sizhe, Haolan Liu, Hamid R. Pourreza, and Hamidreza Mahyar. "Deep learning approaches in pavement distress identification: A review." *arXiv preprint arXiv:2308.00828*(2023).
- [51] Guerrieri, Marco, Giuseppe Parla, Masoud Khanmohamadi, and Larysa Neduzha. 2024. "Asphalt Pavement Damage Detection through Deep Learning Technique and Cost-Effective Equipment: A Case Study in Urban Roads Crossed by Tramway Lines" *Infrastructures* 9, no. 2: 34. <https://doi.org/10.3390/infrastructures9020034>

- [52] Saha, Poonam Kumari, Deeksha Arya, Ashutosh Kumar, Hiroya Maeda, and Yoshihide Sekimoto. "Road rutting detection using deep learning on images." In 2022 IEEE international conference on big data (big data), pp. 1362-1368. IEEE, 2022.
- [53] François Chollet and others, "Keras", 2015, GitHub repository: <https://github.com/keras-team/keras>.
- [54] TensorFlow. (n.d.). Distributed training with TensorFlow. Retrieved from [https://www.tensorflow.org/guide/distributed\\_training](https://www.tensorflow.org/guide/distributed_training)
- [55] Creswell, Antonia, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. "Generative adversarial networks: An overview." IEEE signal processing magazine 35, no. 1 (2018): 53-65.
- [56] Gui, Jie, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. "A review on generative adversarial networks: Algorithms, theory, and applications." IEEE transactions on knowledge and data engineering 35, no. 4 (2021): 3313-3332.
- [57] scikit-image. (2024). Convex hull of a shape — scikit-image 0.24.0 documentation. Retrieved from [https://scikit-image.org/docs/0.24.x/auto\\_examples/edges/plot\\_convex\\_hull.html](https://scikit-image.org/docs/0.24.x/auto_examples/edges/plot_convex_hull.html)
- [58] A. Neubeck and L. Van Gool, "Efficient Non-Maximum Suppression," 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 2006, pp. 850-855, doi: 10.1109/ICPR.2006.479.