Generative Diffusion Models for Agricultural AI: Plant Image Generation, Indoor-to-Outdoor Translation, and Human Preference Alignment

by

Da Tan

A Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
of the University of Manitoba
in partial fulfillment of the requirements of the degree of

Master of Science

Department of Computer Science
University of Manitoba
Winnipeg

Thesis advisor

Christopher Henry

Author

Da Tan

Generative Diffusion Models for Agricultural AI: Plant Image Generation, Indoor-to-Outdoor Translation, and Human Preference Alignment

Abstract

The success of modern agricultural artificial intelligence (AI) depends heavily on access to large-scale, diverse, and annotated plant image datasets. However, collecting such datasets in real-world field conditions is costly, labor-intensive, and constrained by seasonal and environmental variability. This thesis investigates the use of diffusion-based generative modeling to address these challenges through plant image synthesis, indoor-to-outdoor translation, and human preference-aligned fine-tuning.

First, a Stable Diffusion model (SD-1.4) was fine-tuned with curated indoor and outdoor plant imagery to generate realistic, text-conditioned images of canola and soybean plants. Quantitative evaluation using Inception Score (IS) and Fréchet Inception Distance (FID), along with downstream experiments on phenotype classification, demonstrated that synthetic images can effectively augment training data and improve model performance.

Second, we explored image translation to bridge the gap between high-resolution indoor plant datasets and limited outdoor field imagery. By combining DreamBooth-based text inversion with image-guided diffusion, indoor plant structures were pre-

served while environmental contexts such as lighting, soil, and stress conditions were rendered according to outdoor semantics. Translated images were evaluated in a weed detection and classification task using YOLOv8, showing consistent improvements as synthetic data ratios increased.

Finally, a preference-guided fine-tuning framework was developed to align generative outputs with expert judgments of quality and botanical realism. A reward model, trained on manually annotated scores, was integrated into a reward-weighted supervised fine-tuning procedure. The resulting preference-aligned model achieved higher subjective quality and stability, albeit with trade-offs in traditional objective metrics such as FID.

Overall, this work demonstrates that diffusion models can generate, translate, and refine plant images in ways that address data scarcity and domain gaps in agricultural AI. By coupling generative pipelines with expert feedback, this thesis introduces a pathway toward data-efficient and user-centered agricultural machine learning systems.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to begin by expressing my sincere gratitude to my supervisor, Dr. Henry, as well as my committee members, Dr. Jeffrey and Dr. Salehkalaibar, for their invaluable guidance and support. I also thank TerraByte Research Group for the datasets used in this research. In addition, I want to express my thanks to Digital Research Alliance of Canada for the computational resources used in the study. Last but not least, I am deeply grateful to my parents, my significant others, my lab mates, and everyone who has supported me throughout this journey.

This thesis is dedicated to somebody special. You know who you are.

Contributions of Authors

This thesis is the result of collaborative contributions outlined as follows:

Da Tan designed the overall research methodology, implemented all models and experiments, analyzed the results, and wrote the manuscript. Da Tan also prepared all figures, conducted the downstream machine learning evaluations, and integrated feedback from his supervisor and committee members into the final thesis document.

Dr. Christopher Henry supervised the research and provided continuous guidance throughout the research process, including conceptual development, methodological refinement, interpretation of results, and thesis organization. He also contributed to revising and improving the manuscript.

Dr. Michael Beck, Dr. Christopher Bidinosti provided the indoor and outdoor plant image datasets used in this thesis. They also contributed technical insights and feedback during weekly group discussions.

Dr. Rob Gulden offered valuable domain expertise in plant science, providing feedback on the biological realism of generated and translated images.

Dr. Ian Jeffrey and Dr. Sadaf Salehkalaibar served as thesis committee members, offering detailed comments on clarity, methodology, experimental design, and presentation, all of which improved the final document.

Chapter 1

Introduction and Motivation

1.1 Introduction

In recent years, image generation technology has made rapid progress. High-quality synthetic images can now be produced at relatively low cost, often guided directly by natural language prompts. Leading this advancement are diffusion models, such as denoising diffusion probabilistic models (DDPM) [1] and latent diffusion models like Stable Diffusion [2], which have achieved state-of-the-art results in generating diverse, high-resolution images. These models offer more flexibility and higher quality than earlier approaches like generative adversarial networks (GANs) [3].

Despite this progress, the use of diffusion models in scientific fields, especially plant science and agriculture, remains limited. In these areas, high-quality and diverse images are essential for tasks such as phenotyping, disease diagnosis, and crop-performance modeling. However, collecting large, well-annotated outdoor field datasets is time-consuming, expensive, and not as straightforward as scraping images

from the internet for other computer vision tasks. Synthetic image generation offers a promising way to overcome these challenges by providing controlled and customizable alternatives to manual data collection.

This thesis explores how fine-tuned diffusion models can help meet three goals: (1) generating realistic plant images from text descriptions; (2) translating indoor plant images (*e.g.*, from greenhouse environments) into realistic outdoor field scenes; and (3) stabilizing the generative model by aligning it with user preferences through fine-tuning. The overall aim is to narrow the gap between lab-grown and real-world agricultural data, making synthetic images more useful for downstream machine learning applications in plant research.

1.2 Motivation

Plant imaging is constrained by limited access to labeled data and high annotation costs. Indoor greenhouse imaging setups often fail to represent the complexity of outdoor settings such as lighting variation, soil texture, pest damages and plant-weed interactions. This domain gap limits the generalization of computer vision models trained only on indoor datasets of plants [4; 5].

Generative models fine-tuned for plant-specific contexts offer a more flexible alternative. By conditioning on descriptive prompts, these models may generate large and diverse image sets that reflect phenotypic traits, growth stages, and different environments. Furthermore, image-to-image translation from indoor greenhouse to outdoor field domains can serve as a data augmentation tool, which may improve the robustness of downstream models for plant segmentation, detection, and classification

[6; 7].

Our work addresses the current lack of generative frameworks for plant imaging. By building on top of diffusion models' excellent generation quality and controllability, we aim to support plant science research with synthetic data tools that are flexible and domain-relevant.

1.3 Problem Statement

This thesis addresses the lack of high-quality, controllable, and domain-adapted synthetic plant images that can support agricultural research. Specifically, it tackles two key challenges:

1. **Domain-Specific Text-to-Image Generation**: Existing diffusion models are not optimized to generate biologically realistic plant images from scientific text prompts. For example, a prompt like *"a canola leaf showing early signs of chlorosis under drought stress"* often produces irrelevant or distorted outputs without proper fine-tuning. This is mainly due to the mismatch between general-language training data and scientific context.

2. **Indoor-to-Outdoor Image Translation**: Models trained on controlled indoor plant images often fail to generalize to outdoor field conditions due to domain shift. Current unpaired translation methods, such as GAN-based models [8; 9; 10], tend to produce unstable results and often lose fine structural details, which are critical for tasks such as plant disease phenotyping. There is a clear need for a method that can translate images of indoor plants into realistic

outdoor scenes while preserving the accuracy of the plant's appearance.

By addressing these problems, the work aims to produce controllable synthetic datasets that augment plant image data and enhance the robustness of downstream machine learning models used in agriculture. Specifically, this thesis integrates three components into a unified workflow (Figure 1.1). First, we fine-tune a Stable Diffusion model for domain-specific text-to-image generation of crop imagery, and performed classification tasks with the augmented synthetic data. Second, we extend this capability to indoor-to-outdoor image translation, addressing the gap between controlled lab data and variable field conditions, and validate the methodology with a weeds-detection task using both real and translated images. Finally, we further improved the quality of the generated images by incorporating a preference-aligned fine-tuning strategy, where expert feedback is encoded into a reward model to guide the generative process toward more stable outputs. Together, these components establish a cohesive framework for data-efficient and user-aligned generative modeling in agriculture, which will be detailed in Chapters 4 through 6.

1.4 Contributions

This thesis presents a unified framework that advances the use of generative diffusion models for agricultural image synthesis, translation, and expert-aligned refinement. The key contributions are summarized as follows:

1. **Diffusion-based Plant Image Generation:** We develop a text-conditioned diffusion pipeline capable of generating high-fidelity indoor and outdoor plant

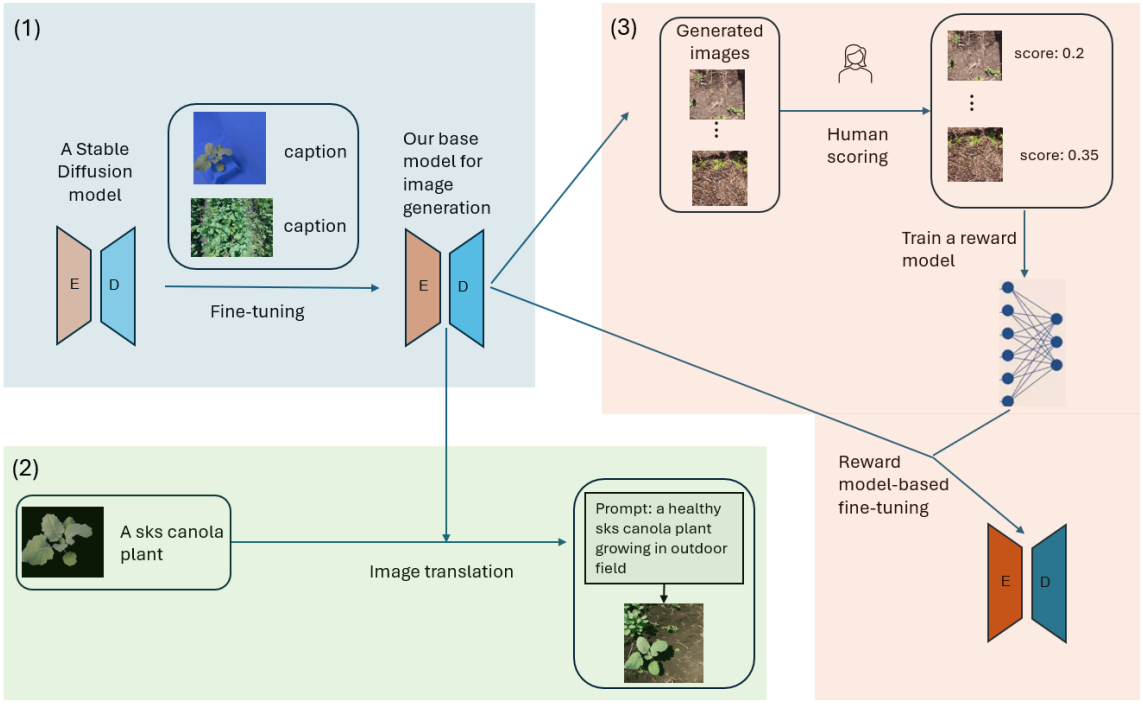

Figure 1.1: Overall workflow of the thesis. The framework integrates three major components as highlighted: (1) text-to-image generation using fine-tuned Stable Diffusion, (2) indoor-to-outdoor image translation via DreamBooth and image-guided diffusion, and (3) preference-aligned fine-tuning with reward modeling. Together, these components provide a cohesive pipeline for generating, adapting, and aligning plant imagery for agricultural applications.

images. Through dataset curation, captioning, and diffusion model fine-tuning, we demonstrate that synthetic images can effectively augment training datasets and improve downstream classification performance.

2. **Indoor-to-Outdoor Image Translation:** We propose both text-guided and image-guided translation approaches that preserve plant structure while adapting lighting, backgrounds, and environmental context. Experiments show that translated images enhance performance in agricultural tasks such as weed detection and classification.

3. **Preference-Guided Diffusion Model Fine-tuning:** We introduce a preference-aligned fine-tuning framework that integrates a reward model and best-of-N selection strategy to align image generation with expert visual preferences. This improves the stability and realism of generated images beyond traditional objective metrics.

1.5 Thesis Structure

The remainder of this thesis is organized to progressively build from theoretical foundations to practical implementation and evaluation. **Chapter 2** reviews related work on synthetic image data generation, diffusion models, and generative AI in agriculture. **Chapter 3** presents the theoretical preliminaries, covering variational autoencoders, diffusion models, latent diffusion, and DreamBooth personalization. **Chapter 4** introduces the text-conditioned plant image generation pipeline, including dataset preparation, captioning, model fine-tuning, evaluation and downstream crop diseases phenotyping. **Chapter 5** extends this work to text-guided and image-guided indoor-to-outdoor translation and examines its benefits for downstream weed detection. **Chapter 6** presents a preference-aligned fine-tuning framework that integrates reward modeling to enhance human-perceived image quality. Finally, **Chapter 7** concludes the thesis. In summary, the thesis follows a coherent progression from theory to application: it begins by establishing the necessary mathematical and conceptual background, then develops and evaluates three generative AI pipelines for agricultural text-to-image generation, image translation, and preference-guided refinement, and concludes with key findings, limitations, and future research opportunities.

Chapter 2

Background and Related Work

2.1 Deep Learning for Image Generation

Image generation is a fundamental problem in computer vision and has seen rapid progress with the advancing of deep learning. The ability to synthesize realistic images from random noise, structured conditions, or textual prompts has not only advanced creative applications but also enabled practical benefits such as data augmentation for downstream machine learning tasks. Early research was dominated by GAN-based models [8], which pioneered adversarial training for realistic image synthesis. Subsequent innovations addressed the limitations of GANs and introduced diffusion-based models [11; 1], which now represent the state-of-the-art in controllable and high-quality image generation. These approaches form the foundation for recent advances in image-to-image translation and editing, which are central to this thesis.

2.1.1 GAN-based Models for Image Generation

GANs [8] have been a cornerstone in the development of deep learning models for image generation. GANs consist of two neural networks, a generator and a discriminator, trained simultaneously in a minimax game: the generator aims to synthesize realistic images, while the discriminator learns to distinguish between real and generated samples. This adversarial training paradigm has been highly effective in producing high-quality images.

Early advances such as the deep convolutional GAN (DCGAN) [12] demonstrated the feasibility of generating natural images with convolutional architectures, providing a strong foundation for subsequent improvements. Conditional GANs (cGANs) [13] extended the framework by conditioning the generator on additional information such as class labels, thereby allowing controlled synthesis of specific image categories.

Further innovations, including progressive growing of GANs (PGGAN) [9], improved stability and enabled the generation of high-resolution images. CycleGAN [10] introduced a novel cycle-consistency loss that enabled unpaired image-to-image translation, facilitating tasks such as translating between different visual domains without paired training data. This capability has been particularly useful in applications where aligned datasets are scarce, such as medical imaging and agricultural plant phenotyping.

2.1.2 Diffusion Models for Image Generation

When introduced in 2015, diffusion models started to emerge as a state-of-art tools in generative modeling. Inspired by nonequilibrium thermodynamics, DDPMs [1]

generate data by reversing a gradual noising process. Unlike GANs, which often suffer from mode collapse and training instability [14], diffusion models achieve improved stability and image diversity by modeling the data distribution more directly through iterative refinements.

A key advancement is the introduction of latent diffusion models (LDMs), which reduce the computational burden of DDPMs by operating in a compressed latent space [2]. This makes high-resolution generation more feasible and scalable. The flexibility of diffusion models also allows conditioning on a variety of modalities, including images, semantic maps, and textual prompts. Recent variants, such as ControlNet [15], further enhance controllability by introducing structural constraints during generation. In ControlNet, an additional network branch receives a structural input, such as an edge map, segmentation mask, or depth map, and this branch guides the denoising process so that the generated image preserves the spatial layout, or object shapes encoded in the input.

2.1.3 Text-to-Image Synthesis

Text-to-image generation is a task where a model generates images from natural language descriptions, enabling controllable content creation. This capability is a key aspect to models such as DALL·E [16], Imagen [17], and Stable Diffusion [2], which leverage pretrained text encoders like CLIP [18] to embed semantic meaning from prompts. The use of large-scale vision-language paired-data for training allows these models to generalize across diverse prompts, and achieve strong alignment between text and image content. However, their performance in out-of-training domains is

limited by the nature of the training data. For example in plant science, prompts describing botanical attributes (*e.g.*, "canola with yellowing leaves under drought stress") often fall outside the distribution of the training corpus, leading to artifacts, inaccuracies, or out-of-domain features in the generated images.

To address this, domain-adaptive fine-tuning techniques, such as DreamBooth [19] or LoRA [20], have been introduced to specialize large generative models using small, focused datasets. These methods preserve the core generative capacity while adapting to new concepts, styles, or domains. In this work, we fine-tune a latent diffusion model using curated plant image datasets. This enables the model to synthesize realistic plant images guided by text prompts specific to agricultural research.

2.1.4 Alternative Generative Models for Text-to-Image Synthesis

While diffusion models have recently become the dominant approach for text-to-image generation, several alternative and complementary techniques have been proposed to enhance or control generative processes. These methods offer different trade-offs in terms of data requirements, training complexity, and generalization. Textual inversion [21] enables personalization of generative models by learning pseudo-token embeddings (*e.g.*, "plantX") that capture the visual identity of a new concept using only a few reference images. This method fine-tunes only the text encoder, leaving the image generator unchanged. It is useful for integrating rare or unseen objects into a pretrained model without full retraining. Another text inversion-based approach, plug-and-play [22], proposes a training-free framework for text-guided image-to-image

translation that uses a pre-trained text-to-image diffusion model. By directly injecting spatial features from a guidance image into the generation process, this method enables fine-grained control over structure while modifying the content according to a target text prompt. Another example comes from SDEdit [23], which allows localized or guided edits to generated images without full re-sampling. This approach modifies intermediate representations (*e.g.*, cross-attention maps) to steer generation toward new prompts or user constraints. More recently, Shi *et. al.* introduced InstantBooth [24], a fast and scalable method for text-guided image personalization that eliminates the need for test-time finetuning. It encodes input images into textual tokens using a learnable image encoder and preserves fine visual details through lightweight adapter layers integrated into a pre-trained text-to-image model, achieving high-quality results faster than prior approaches.

2.2 Deep Learning for Image-to-Image Translation

Image-to-image translation has emerged as an important task in computer vision, aiming to learn mappings between two visual domains while preserving semantic structure. Typical applications include colorization, style transfer, semantic-to-realistic image conversion, and adapting across environmental domains (*e.g.*, plant images from indoor to outdoor scenes). The field has evolved through several stages of methodological development. Early approaches relied on supervised translation with paired training data, enabling direct mappings between aligned domains. To address

the difficulty of obtaining such pairs, unsupervised translation frameworks were introduced, learning correspondences from unaligned data. More recently, diffusion-based translation has gained prominence, providing improved stability and image fidelity over GAN-based methods. In parallel, specialized advances in image editing and inpainting have extended translation frameworks to localized transformations, offering fine-grained control of image content. The following subsections review each of these directions in turn.

2.2.1 Supervised Image Translation

Image-to-image translation refers to learning a mapping between two visual domains, such as transforming grayscale images to color. Classical supervised approaches rely on paired datasets, where corresponding images from source and target domains are available. For example, Pix2Pix [25] introduced a conditional GAN framework that learns mappings from paired training data, producing high-quality outputs across a variety of tasks such as semantic label to photo translation and image colorization. While effective, these models are constrained by the need for aligned data, which is costly and often infeasible to obtain in many real-world applications.

2.2.2 Unsupervised Image Translation

To overcome the scarcity of paired data, unsupervised approaches have been developed. CycleGAN [10] introduced the concept of cycle-consistency loss to enforce that an image translated from source to target and back should recover the original. This framework enabled translation across unaligned domains such as horses and ze-

bras, Monet paintings and photographs, and seasonal landscape changes. Subsequent work such as DualGAN [26] and UNIT [27] further advanced this direction, demonstrating strong results in style transfer. These methods significantly expanded the applicability of image translation but still faced challenges with training stability and mode collapse, common to GAN-based frameworks.

2.2.3 Diffusion-Based Image Translation

Recent advances in diffusion models have provided new opportunities for image-to-image translation, offering improved image quality and more stable training compared to GAN-based methods. Palette [28] formulates image translation as a conditional diffusion process, enabling tasks such as colorization, inpainting, and segmentation map translation with impressive fidelity. SDEdit [23] leverages stochastic differential equations to guide image editing and translation by injecting controlled noise into the input and denoising under conditional guidance.

2.2.4 Image Editing and Inpainting

Beyond full-domain translation, image-to-image frameworks have also been extended to localized editing and inpainting tasks. These approaches allow modifying or filling specific image regions while maintaining global coherence. Classical methods like DeepFill [29] employed GANs with attention mechanisms to generate realistic inpainted content. More recently, diffusion-based inpainting methods such as those in SD pipelines [2] have enabled high-quality editing by conditioning on masked regions and text prompts. These advances expand the scope of image translation beyond

rigid source-to-target mappings, enabling more flexible and controllable editing.

2.3 Supervised Image Refining with Human Feedback

Pre-trained generative models can produce visually realistic images, yet the outputs often fail to capture the subtle semantic or aesthetic qualities valued by end-users. This has motivated the development of methods that incorporate expert feedback or preference signals to better align model outputs with user expectations. In this section, we review the most relevant approaches including reinforcement learning, differentiable ranking, and best-of-N sample–based fine-tuning that refine image generation through preference-guided optimization.

2.3.1 Reinforcement Learning with Human Feedback

One of the most widely adopted paradigms for preference alignment is RLHF. Originating from natural language processing [30; 31], reinforcement learning with human feedback (RLHF) has been successfully adapted to diffusion models. For example, Black et al. [32] applied proximal policy optimization (PPO) to fine-tune diffusion models for aesthetic preferences, demonstrating improvements over supervised baselines. Similarly, Lee *et al.* [33] explored preference-conditioned diffusion, showing that reinforcement signals can guide image generation toward user-specified quality dimensions. While effective, RLHF is computationally expensive and often unstable when applied to high-dimensional image data, motivating exploration of more direct

and stable alternatives.

2.3.2 Differentiable Ranking and Direct Preference Optimization

To alleviate the instability of full reinforcement learning pipelines, differentiable ranking and direct preference optimization have been proposed. DRaFT (Differentiable Ranking Fine-Tuning) [34] directly optimizes a diffusion model by backpropagating through a differentiable ranking objective, allowing gradient-based updates toward preferred generations without the complexity of policy optimization.

2.3.3 Best-of-N Selection and Supervised Fine-Tuning

A simpler but highly practical approach is to exploit rejection sampling and supervised fine-tuning on preferred subsets of generated data, this is a paradigm often referred to as best-of-N selection. In this framework, the model generates N candidates per prompt, a reward model or human ranks the outputs, and the top-k candidates are adopted as pseudo-targets for supervised fine-tuning. Recently, preference optimization approaches such as direct preference optimization (DPO) [35] have gained traction in the language domain and inspired adaptations for vision models, where pairwise preferences are used to directly adjust the model distribution toward user-preferred samples. This best-of-N strategy has been employed in more recent works on diffusion alignment such as [36], and is theoretically connected to policy optimization frameworks such as PPO via rejection-based fine-tuning [37]. Compared to full RL approaches, this method trades off some theoretical optimality for greater sim-

plicity, leveraging the natural ability of diffusion models to generate multiple diverse samples per prompt.

2.4 Generative Methods for Agricultural Applications

Synthetic data plays a crucial role in data augmentation, particularly in scientific domains that face data scarcity and class imbalance. In agriculture, collecting high-quality, annotated images of crops, weeds, or diseased plants under field conditions is expensive, labor-intensive, and often limited by seasonal or environmental constraints [38].

Early applications of deep learning in agriculture predominantly focused on image-based analysis tasks. The majority of studies centered on image classification and object identification, aiming to recognize crops, detect obstacles in the field [39; 40], or count fruits from visual imagery [41; 42]. Beyond visual recognition, a smaller number of works explored predictive modeling, such as forecasting crop yield [43], estimating soil moisture content [44], and predicting weather parameters [45]. In terms of application domains, the majority of studies targeted crop-related problems, while the remaining efforts were distributed across weed detection, land-cover classification, soil property estimation, livestock monitoring, obstacle detection, and weather prediction [38]. These secondary areas were comparatively underexplored, reflecting an early-stage research trend in which deep learning methods were primarily used for visual perception and classification tasks.

With the rise of deep generative models, researchers began applying GANs to agricultural imagery. For example, several studies synthesize or translate plant images to enrich datasets [6; 46]. However, GANs have limitations in capturing fine-grained variation, and they lack the fine semantic controllability offered by diffusion-based or text-conditioned models [14; 2].

Broad reviews of generative augmentation in farming underscore both promise and challenges [47; 48]. One IEEE review of GAN-based augmentation in farming highlights its adoption for tasks such as disease detection, weed removal, and yield prediction, while also noting difficulties in model stability, mode collapse, and domain generalization [47]. Another review [48] emphasizes that many generative approaches in agriculture remain at early stages, constrained by small-scale datasets, narrow crop domains, and limited evaluation of downstream utility (*e.g.*, for classification or detection).

In sum, while prior work has begun to leverage GANs for agricultural image synthesis, challenges remain in realism, diversity, and control. In contrast, this thesis situates diffusion-based models (with text and translation conditioning) as a more flexible and high-fidelity alternative, capable not only of generating and translating realistic plant images but also of aligning with downstream tasks and human preference signals.

Chapter 3

Theoretical Preliminaries

This chapter outlines the theoretical foundations underlying the proposed generative frameworks used throughout this thesis. Modern diffusion-based models build upon two key principles: variational inference and iterative denoising. Variational autoencoders provide a probabilistic latent representation that enables efficient encoding and reconstruction of high-dimensional data, while diffusion models learn to generate realistic samples by reversing a gradual noising process. Latent diffusion models combine these two paradigms by performing the diffusion process in a learned latent space rather than directly in pixel space, achieving high-quality synthesis at reduced computational cost. Finally, DreamBooth extends this framework for subject-specific fine-tuning, enabling personalized generation and translation of plant images with minimal data.

3.1 Variational Autoencoders

Variational autoencoders (VAEs) [49] are generative models that combine deep neural networks with variational inference to learn latent variable representations of data. They provide a probabilistic framework for encoding data into a latent space and decoding samples back into the input domain, forming a key theoretical foundation for latent diffusion models. Throughout this section, we follow the notational conventions and VAE formulation presented in *Understanding Deep Learning* [50].

In our framework, we adopted a VAE rather than a standard Autoencoder (AE) [51; 52] because VAEs provide a continuous Guassian-like latent space that is essential for image generation via sampling. While an AE simply learns to map images to arbitrary latent codes deterministically, it does not impose any constraints on the distribution of the latent space, which often lead to disconnected latent regions that prevent from sampling and interpolation. Input of random vectors in the latent space therefore will result in meaningless output images. In contrast, the VAE regularizes the latent space by enforcing a distribution shape through a KL divergence term [53]. This constraint results in a smooth latent space where new samples can be drawn by random sampling, and sampling interpolation will generate similar images. Additionally, the probabilistic nature of VAEs enables output variations and prevents the model from memorizing training data. These features make VAEs more suitable than AEs as the latent foundation for our latent diffusion model.

3.1.1 Latent Variable Models

Latent variable models are probabilistic frameworks that assume the observed data $\mathbf{x}$ is reconstructed from an underlying set of latent variables $\mathbf{z}$. These latent variables capture the hidden structure or semantics of the data, providing a compact and continuous representation that enables generation. The generative process is formalized through the marginal likelihood:

$$p_\phi(\mathbf{x}) = \int p_\phi(\mathbf{x} \mid \mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}, \tag{3.1}$$

where $p_\phi(\mathbf{x} \mid \mathbf{z})$ is the likelihood function (decoder) parameterized by neural network parameters ϕ, and $p(\mathbf{z})$ is a prior distribution, typically chosen as a standard normal $\mathcal{N}(0, \mathbf{I})$. This integral expresses how the data distribution arises from all possible configurations of latent variables, weighted by their prior probabilities.

In essence, latent variable models define two key directions:

- **Inference:** Estimating $\mathbf{z}$ given $\mathbf{x}$ by approximating a latent inference model $q_\theta(\mathbf{z} \mid \mathbf{x})$, which enables feature extraction or encoding.
- **Generation:** Sampling $\mathbf{z} \sim p(\mathbf{z})$ and decoding $\mathbf{x} \sim p_\phi(\mathbf{x} \mid \mathbf{z})$, which allows new data synthesis.

This leads to the structure of the VAE, which learns both a generative model $p_\phi(\mathbf{x} \mid \mathbf{z})$ and an approximate inference model $q_\theta(\mathbf{z} \mid \mathbf{x})$ using neural networks trained jointly under a probabilistic objective.

3.1.2 Variational Approximation and the Evidence Lower Bound

To train the VAE model, we aim to maximize the log-likelihood of the observed data $\{\mathbf{x}_i\}_{i=1}^N$ with respect to the model parameters ϕ:

$$\hat{\phi} = \arg\max_{\phi} \left[\sum_{i=1}^{N} \log p_\phi(\mathbf{x}_i) \right], \tag{3.2}$$

where the likelihood of each data point is defined as

$$p_\phi(\mathbf{x}_i) = \int p_\phi(x_i|z) \cdot p(z) d\mathbf{z}. \tag{3.3}$$

The term $p_\phi(\mathbf{x})$, often called the evidence, requires integration over all possible latent variables and is computationally intractable. Instead of evaluating this term directly, we aim to maximize it indirectly. Starting from the marginal log-likelihood:

$$\log p_\phi(\mathbf{x}) = \log \int p_\phi(\mathbf{x}, \mathbf{z})\, d\mathbf{z}, \tag{3.4}$$

we introduce the latent inference distribution $q_\theta(\mathbf{z})$ to approximate the true posterior. By multiplying and dividing by $q_\theta(\mathbf{z})$ inside the integral and applying Jensen's inequality, we obtain:

$$\log p_\phi(\mathbf{x}) = \log \int q_\theta(\mathbf{z}) \frac{p_\phi(\mathbf{x}, \mathbf{z})}{q_\theta(\mathbf{z})}\, d\mathbf{z} \tag{3.5}$$

$$\geq \int q_\theta(\mathbf{z}) \log \frac{p_\phi(\mathbf{x}, \mathbf{z})}{q_\theta(\mathbf{z})}\, d\mathbf{z}. \tag{3.6}$$

The right-hand side defines the evidence lower bound (ELBO):

$$\mathcal{L}_{\text{ELBO}}(\theta, \phi) = \mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[\log p_\phi(\mathbf{x}, \mathbf{z}) - \log q_\theta(\mathbf{z})]\,. \tag{3.7}$$

Expanding the joint term $p_\phi(\mathbf{x}, \mathbf{z}) = p_\phi(\mathbf{x} \mid \mathbf{z})p(\mathbf{z})$, the ELBO can be rewritten as:

$$\mathcal{L}_{\text{ELBO}}(\theta, \phi) = \mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[\log p_\phi(\mathbf{x})] - \text{KL}(q_\theta(\mathbf{z}) \,\|\, p(\mathbf{z}))\,. \tag{3.8}$$

Since the KL term is non-negative, the ELBO indeed provides a lower bound on the data likelihood. This formulation justifies the name "evidence lower bound," as it bounds the evidence $\log p_\phi(\mathbf{x})$ from below. And this decomposition provides an intuitive interpretation:

- The first term, $\mathbb{E}_{q_\theta}[\log p_\phi(\mathbf{x})]$, represents the reconstruction accuracy, which encourages the decoder to generate data close to the input.
- The second KL term acts as a regularization penalty, which ensures the latent distribution remains close to the prior $p(\mathbf{z})$.

Thus, the ELBO can be interpreted as a trade-off between reconstruction quality and latent regularity:

$$\mathcal{L}_{\text{ELBO}} = \text{Reconstruction Loss} - \text{KL Divergence}. \tag{3.9}$$

In practice, the reconstruction loss is often implemented as a mean squared error for continuous data like images, while the KL divergence can be computed when both $q_\theta(\mathbf{z} \mid \mathbf{x})$ and $p(\mathbf{z})$ are Gaussian. By maximizing this objective, the model learns both a meaningful latent representation and a generative process capable of synthesizing realistic samples.

3.1.3 The Variational Autoencoder and Reparameterization Trick

The VAE [49] operates variational inference using deep neural networks. As mentioned in earlier sections, VAE consists of two key components: an encoder network

that parameterizes the approximate posterior distribution $q_\theta(\mathbf{z} \mid \mathbf{x})$, and a decoder network that models the likelihood $p_\phi(\mathbf{x} \mid \mathbf{z})$. Together, they learn to encode input data into a continuous latent space and decode samples from this latent space to reconstruct the original data. The VAE is optimized by maximizing the ELBO as defined in Equation 3.8. An overview of its architectural components is illustrated in Figure 3.1.

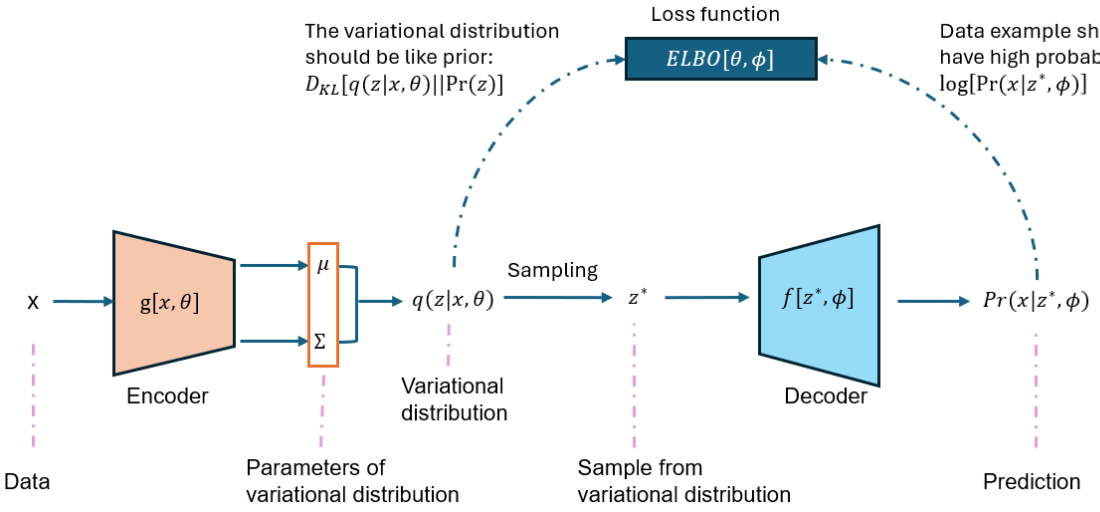

Figure 3.1: Variational Autoencoder architecture, reproduced and adapted from *Understanding Deep Learning* [50], p. 337. The encoder $g[\mathbf{x}, \theta]$ maps a training example $\mathbf{x}$ to the parameters $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ of the variational distribution $q(\mathbf{z} \mid \mathbf{x}, \theta)$. A latent vector $\mathbf{z}$ is then sampled from this distribution and passed to the decoder $f[\mathbf{z}, \phi]$ to reconstruct the data $\mathbf{x}$. The model is trained by minimizing the negative evidence lower bound, which balances the reconstruction accuracy of $\mathbf{x}$ and the regularization term that enforces similarity between $q(\mathbf{z} \mid \mathbf{x}, \theta)$ and the prior $p(\mathbf{z})$.

Reparameterization Trick. A challenge in training VAEs is that direct sampling from $q_\theta(\mathbf{z} \mid \mathbf{x})$ prevents gradients from propagating through stochastic nodes. The reparameterization trick resolves this by expressing $\mathbf{z}$ as a deterministic function of

the encoder outputs and an auxiliary random variable:

$$\mathbf{z} = \boldsymbol{\mu}_\theta(\mathbf{x}) + \boldsymbol{\sigma}_\theta(\mathbf{x})\boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}). \tag{3.10}$$

This transformation makes the sampling process differentiable with respect to θ, enabling standard backpropagation.

Training Objective. The overall training objective is to maximize the ELBO, equivalently minimizing the negative ELBO loss:

$$\mathcal{L}_{\text{VAE}}(\theta, \phi) = \underbrace{\mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[-\log p_\phi(\mathbf{x} \mid \mathbf{z})]}_{\text{Reconstruction Loss}} + \underbrace{\text{KL}\big(q_\theta(\mathbf{z} \mid \mathbf{x}) \,\|\, p(\mathbf{z})\big)}_{\text{KL Regularization}}. \tag{3.11}$$

The first term encourages accurate reconstruction of the data, while the second enforces latent regularization by keeping the learned posterior close to the prior distribution, usually a standard normal $\mathcal{N}(0, \mathbf{I})$.

Implementation. In practice, both the encoder and decoder are parameterized by CNNs for image data, allowing the model to capture spatial hierarchies and fine-grained texture information. Once trained, the decoder can be used as a generative model: sampling $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ and decoding through $f_\phi(\mathbf{z})$ produces new, realistic images.

3.2 Foundations of Diffusion Models

Diffusion models constitute a class of generative models that learn to synthesize data by reversing a gradual noising process. They have recently emerged as a powerful alternative to GANs, offering improved training stability and state-of-the-art

performance in high-fidelity image generation [1; 54]. This section introduces the theoretical foundations of diffusion models, including the convolutional neural network backbone, the U-Net architecture, and the mathematical formulation of the forward and reverse diffusion processes that underpin the DDPM.

3.2.1 Convolutional Neural Networks and the U-Net Architecture

Convolutional neural networks (CNNs) [55] form the structural foundation of modern diffusion-based architectures. Their capacity to capture hierarchical spatial correlations through localized convolutional kernels makes them particularly effective for modeling visual data, where both low-level textures and high-level semantics must be represented. Building upon this principle, the U-Net architecture [56] extends the CNN design by introducing an encoder–decoder framework with symmetric skip connections, as illustrated in Figure 3.2.

In diffusion models, the U-Net serves as the backbone of the denoising network that predicts and removes noise from corrupted images at each diffusion step. The encoder progressively downsamples the noisy input, capturing high-level semantic and contextual information by increasing feature channels while reducing spatial resolution. The decoder performs the inverse operation, upsampling the latent features to reconstruct fine image details. Skip connections directly link corresponding encoder and decoder layers, enabling the network to transfer fine-grained spatial information, such as texture and edge details, lost during downsampling.

Through this hierarchical structure, the U-Net effectively combines global context

from deeper layers with local precision from shallower ones, allowing the model to iteratively refine image representations during the denoising process. A single U-Net is reused across all diffusion timesteps, conditioned by a sinusoidal time embedding injected into each block to guide the denoising at different noise levels.

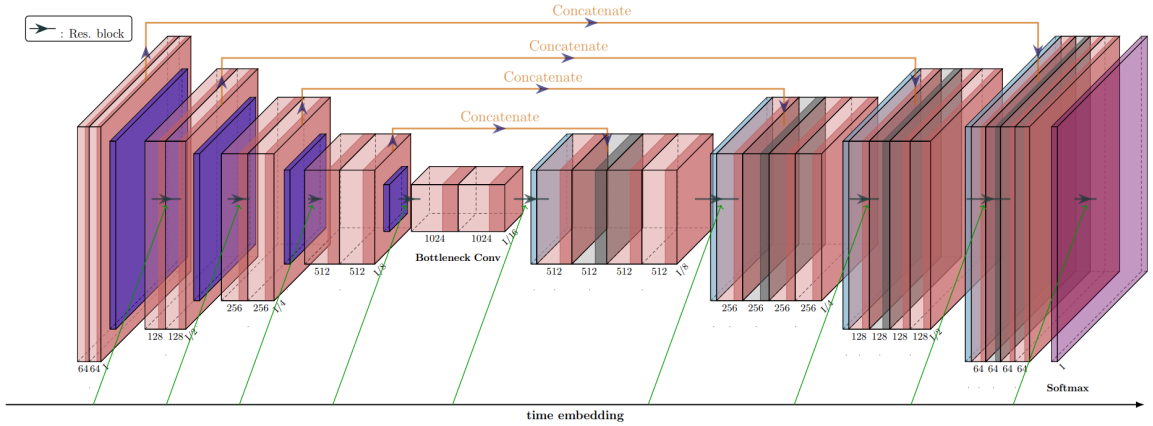

Figure 3.2: U-Net architecture used in diffusion models for images. The network predicts the noise added to the image at each diffusion step. It comprises an encoder that progressively reduces spatial resolution while increasing the number of feature channels, and a decoder that performs the inverse operation, restoring resolution while reducing channels. Encoder feature maps are concatenated with their corresponding decoder counterparts through skip connections. Adjacent layers are connected by residual blocks. A single network is reused across all time steps by injecting a sinusoidal time embedding, processed by a shallow neural network, into the channels at every stage of the U-Net.

3.2.2 Encoder (Forward Process)

In diffusion-based generative modeling, the encoder corresponds to the forward diffusion process, which progressively transforms an input image $\mathbf{x}$ into a sequence of

noisy latent representations $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_T$ through stochastic transitions:

$$\begin{aligned} \mathbf{z}_1 &= \sqrt{1-\beta_1}\,\mathbf{x} + \sqrt{\beta_1}\,\boldsymbol{\epsilon}_1, \\ \mathbf{z}_t &= \sqrt{1-\beta_t}\,\mathbf{z}_{t-1} + \sqrt{\beta_t}\,\boldsymbol{\epsilon}_t, \quad \forall\, t \in \{2, \ldots, T\}, \end{aligned} \tag{3.12}$$

where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$ is Gaussian noise drawn independently at each step, and $\beta_t \in [0, 1]$ controls the noise variance at timestep t. The first term in each equation attenuates the contribution from the previous state, while the second term introduces additional Gaussian perturbation. The sequence $\{\beta_t\}_{t=1}^{T}$ defines the noise schedule, determining how quickly the signal is blended with noise over time. Unlike conventional neural encoders, this process is fixed with no learnable parameters. At each timestep t, Gaussian noise is incrementally added according to $\{\beta_t\}_{t=1}^{T}$, gradually converting the structured input $\mathbf{x}$ into nearly pure noise $\mathbf{z}_T$.

The encoder performs this transformation via a sequence of Gaussian perturbations governed by the noise schedule $\{\beta_t\}_{t=1}^{T}$. Formally, the forward process defines a Markov chain of latent variables $\{\mathbf{z}_t\}_{t=1}^{T}$ as:

$$q(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = \mathcal{N}\Big(\mathbf{z}_t; \sqrt{1-\beta_t}\,\mathbf{z}_{t-1},\, \beta_t \mathbf{I}\Big)\,. \tag{3.13}$$

Starting from $\mathbf{z}_0 = \mathbf{x}$, the clean data sample, the encoder gradually destroys the signal content of the image over T steps, such that $\mathbf{z}_T$ approaches an isotropic Gaussian distribution $\mathcal{N}(0, \mathbf{I})$. A closed-form expression for directly sampling $\mathbf{z}_t$ from $\mathbf{x}$ is:

$$q(\mathbf{z}_t \mid \mathbf{x}) = \mathcal{N}\big(\mathbf{z}_t; \sqrt{\bar{\alpha}_t}\,\mathbf{x},\, (1-\bar{\alpha}_t)\mathbf{I}\big)\,, \tag{3.14}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$ is the cumulative product of the noise schedule parameters.

This closed form enables efficient sampling at arbitrary timesteps without explicitly simulating all intermediate steps.

Intuitively, each step in the forward process can be viewed as encoding higher levels of uncertainty and abstracting the data representation into a progressively noisier latent space. By the end of the process ($t = T$), the latent variable $\mathbf{z}_T$ retains no observable structure of the original image, serving as a compressed and randomized latent encoding.

The role of the reverse process (decoder), described in the next subsection, is to learn how to invert this stochastic encoding by reconstructing $\mathbf{x}$ from $\mathbf{z}_T$ through a learned denoising process.

3.2.3 Decoder (Reverse Process)

The decoder in a diffusion model corresponds to the reverse denoising process, which reconstructs realistic images by inverting the forward (encoding) diffusion process. Starting from a noise sample $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$, the decoder learns to iteratively remove noise and recover a clean data sample $\mathbf{x}$. This process can be interpreted as learning a series of probabilistic mappings from the latent variable $\mathbf{z}_T$ to $\mathbf{z}_{T-1}$ to $\mathbf{z}_{T-2}$ and so on, until we reach the data $\mathbf{z_0} = \mathbf{x}$. Formally, the reverse process is modeled as another Markov chain parameterized by a neural network with learnable parameters θ:

$$p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t) = \mathcal{N}(\mathbf{z}_{t-1};\, \boldsymbol{\mu}_\theta(\mathbf{z}_t, t),\, \boldsymbol{\Sigma}_\theta(\mathbf{z}_t, t))\,, \tag{3.15}$$

where $\boldsymbol{\mu}_\theta$ and $\boldsymbol{\Sigma}_\theta$ represent the predicted mean and variance of the denoised estimate at each timestep. The decoder is typically implemented as a U-Net conditioned on the timestep t and the input prompt embedding in text-to-image models.

Because the true reverse conditional distribution $q(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$ is analytically intractable, the neural network approximates its mean by predicting the added Gaussian noise $\boldsymbol{\epsilon}_t$ used in the forward process. Following the formulation of Ho *et al.* [1], the mean of the reverse distribution can be expressed as:

$$\boldsymbol{\mu}_\theta(\mathbf{z}_t, t) = \frac{1}{\sqrt{1-\beta_t}} \left(\mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \right), \tag{3.16}$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t)$ is the model's predicted noise at step t. Given this, the decoder reconstructs the previous latent variable $\mathbf{z}_{t-1}$ using a sampled Gaussian perturbation:

$$\mathbf{z}_{t-1} = \boldsymbol{\mu}_\theta(\mathbf{z}_t, t) + \boldsymbol{\Sigma}_\theta^{1/2}(\mathbf{z}_t, t)\, \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}). \tag{3.17}$$

Through this iterative denoising, the decoder transforms a sample of pure noise $\mathbf{z}_T$ into an image $\mathbf{x}$ that follows the learned data distribution. In practice, this reverse process is implemented as a sequence of T neural function evaluations, each refining the intermediate latent representation by predicting and removing a portion of the noise. During inference, the denoising trajectory can often be accelerated using fewer timesteps (e.g., 20–50 steps) with improved sampling methods [57; 54], greatly reducing computation.

Intuitively, the decoder acts as the generative core of the model: while the encoder progressively destroys structure to obtain a simple Gaussian prior, the decoder learns to reconstruct structured, semantically meaningful data from that prior distribution.

When conditioned on textual or multimodal embeddings, this process becomes a controllable generator capable of synthesizing images aligned with human-provided prompts.

3.2.4 Training Objective and Loss Function

Training a diffusion model involves learning the parameters θ of the decoder network so that the reverse process accurately reconstructs clean data samples from noisy latent variables. The goal is to approximate the true reverse conditional distribution $q(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$ with a parameterized model $p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$ by minimizing the divergence between them. This can be achieved by optimizing a variational lower bound on the log-likelihood of the data distribution $p_\theta(\mathbf{x})$:

$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_q \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z}_{1:T})}{q(\mathbf{z}_{1:T} \mid \mathbf{x})} \right]. \tag{3.18}$$

Ho *et al.* [1] showed that this bound can be simplified to a denoising objective where the model learns to predict the Gaussian noise $\boldsymbol{\epsilon}$ that was added to $\mathbf{x}$ during the forward process. Specifically, the loss function becomes:

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{t,\mathbf{x},\boldsymbol{\epsilon}} \left[\left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \right\|^2 \right], \tag{3.19}$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t)$ denotes the model's predicted noise at timestep t. Intuitively, the model learns to reverse the corruption process by estimating the exact noise added at each step. Once this noise is predicted accurately, the network can recover the underlying clean data sample $\mathbf{x}$ by inverting the forward diffusion equations.

3.2.5 Implementation Algorithm

The training and inference of diffusion models follow a structured iterative procedure, which alternates between adding and removing Gaussian noise. The overall workflow consists of two primary stages: (1) the forward diffusion training process, where the model learns to predict added noise, and (2) the reverse sampling process, where the model generates new data by progressively denoising random noise. Both procedures are outlined below.

Algorithm 1 Diffusion Model Training

Require: Training dataset $\mathbf{x}$, noise schedule $\{\beta_t\}_{t=1}^{T}$

Ensure: Trained model parameters θ

repeat
  for each mini-batch $\mathbf{x}_i \in \mathcal{B}$ **do**
    Sample timestep $t \sim \text{Uniform}\{1, \ldots, T\}$
    Sample noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$
    Compute noised sample: $\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_i + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}$
    Compute loss: $\mathcal{L}_i = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t)\|^2$
  Accumulate losses and update θ via gradient descent
until convergence

Algorithm 2 Sampling from the Trained Diffusion Model

Require: Trained model $\boldsymbol{\epsilon}_\theta$, noise schedule $\{\beta_t\}_{t=1}^{T}$

Ensure: Generated image sample $\mathbf{x}$

1: Sample initial noise $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$

2: **for** $t = T, T-1, \ldots, 2$ **do**

3: Predict mean: $\boldsymbol{\mu}_\theta(\mathbf{z}_t, t) = \frac{1}{\sqrt{1-\beta_t}} \left(\mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t) \right)$

4: Sample noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$

5: Update latent: $\mathbf{z}_{t-1} = \boldsymbol{\mu}_\theta(\mathbf{z}_t, t) + \sigma_t \boldsymbol{\epsilon}$

6: **return** Reconstructed sample $\mathbf{x} = \mathbf{z}_0$

In the training phase (Algorithm 1), the model learns to predict the Gaussian noise added to each data sample across randomly selected diffusion timesteps. The optimization minimizes the mean-squared error between the predicted and true noise values, enabling the network to learn the conditional denoising distribution $p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$.

During inference (Algorithm 2), image generation begins from a sample of pure Gaussian noise $\mathbf{z}_T$. The trained model is applied iteratively in reverse order of timesteps, producing progressively less noisy images until the final synthetic image $\mathbf{z}_0$ is obtained. This iterative denoising can be interpreted as a learned generative process that maps a simple prior into complex visual data.

3.3 Latent Diffusion Models

Latent Diffusion Models (LDMs) [2] integrate the strengths of VAEs and diffusion models into a unified generative framework. While diffusion models excel at high-fidelity image synthesis through iterative denoising, their operation directly in pixel space is computationally expensive. LDMs address this limitation by performing the diffusion process in a compact latent space learned by a VAE. The VAE encoder compresses the image into a lower-dimensional latent representation that preserves perceptually meaningful features, while the diffusion model learns to denoise within this space. This combination enables efficient training and inference without compromising visual quality, forming the backbone of many modern image generative models, including Stable Diffusion.

3.3.1 Model Architecture and Latent Space

The key insight of LDMs is to introduce an autoencoder that compresses the high-dimensional image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$ into a lower-dimensional latent representation $\mathbf{z} \in \mathbb{R}^{h \times w \times c}$, where $h \ll H$ and $w \ll W$. The autoencoder consists of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, such that:

$$\mathbf{z} = \mathcal{E}(\mathbf{x}), \quad \hat{\mathbf{x}} = \mathcal{D}(\mathbf{z}). \tag{3.20}$$

The latent space $\mathbf{z}$ captures the semantic and structural information of the image while discarding pixel-level redundancy. Diffusion is then applied in this compact latent space, which drastically reduces the computational overhead and memory requirements of training.

3.3.2 Connection between VAE and Latent Diffusion

The latent autoencoding stage of LDMs is conceptually rooted in VAEs [49; 58]. The encoder $\mathcal{E}$ and decoder $\mathcal{D}$ are trained with a reconstruction loss and a KL divergence regularization, as in standard VAEs. Once the latent space is learned, the diffusion model operates directly on $\mathbf{z}$ instead of $\mathbf{x}$. This integration ensures that the latent space is both expressive enough to preserve semantic details and structured enough to support stable diffusion processes. Thus, LDMs can be viewed as a two-stage generative pipeline: (1) a VAE for learning a compact latent representation, and (2) a diffusion model for generative sampling in this latent domain.

3.3.3 Diffusion in Latent Space and Denoising Objective

Instead of applying noise directly to pixels, LDMs apply the forward and reverse diffusion processes to latent variables $\mathbf{z}_t$. The forward process adds Gaussian noise:

$$q(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = \mathcal{N}\left(\mathbf{z}_t; \sqrt{1-\beta_t}\mathbf{z}_{t-1}, \beta_t \mathbf{I}\right), \tag{3.21}$$

while the reverse process is parameterized by a neural network ϵ_θ, which predicts the noise added at each step. Similar to diffusion models, the training objective of LDMs reduces to predicting the noise ϵ added to the latent representation $\mathbf{z}_t$:

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{t,\mathbf{z}_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t, y)\|^2\right], \tag{3.22}$$

where $\mathbf{z}_0$ is the clean latent code, $\epsilon \sim \mathcal{N}(0, I)$, and y denotes optional conditioning information such as text embeddings. The decoder $\mathcal{D}$ then maps the final latent $\mathbf{z}_0$ back into the pixel domain to obtain the generated image.

3.3.4 Text-Conditioning in the Denoising Process

An innovation of LDMs [2] lies in their ability to condition the denoising process on textual descriptions. This conditioning mechanism enables semantic control over the generation process by aligning linguistic information with visual representations at each denoising step. This is achieved by injecting text-derived embeddings into the U-Net denoiser through cross-attention layers, as illustrated in Figure 3.3

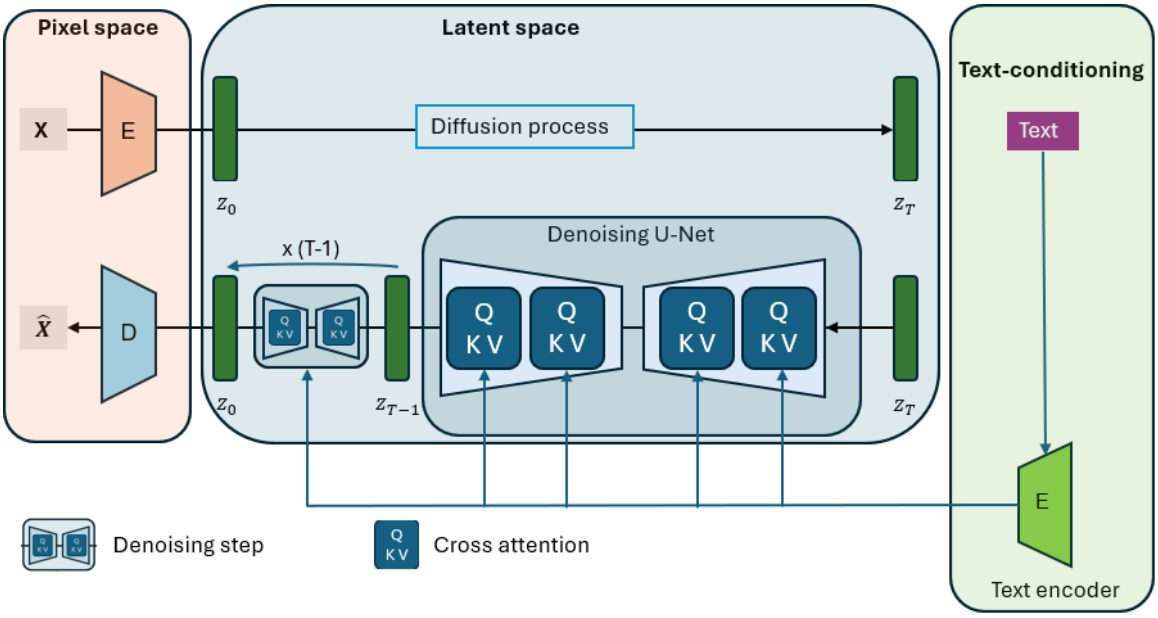

Figure 3.3: **Cross-attention mechanism in text-conditioned denoising.** The U-Net denoiser operates on noisy latent features $\mathbf{z}_t$ at each timestep t. Intermediate image feature maps are projected into query vectors Q, which attend over key (K) and value (V) projections of text embeddings derived from a transformer-based text encoder.

Text Embedding Representation

Given a text prompt y, a pre-trained language model such as CLIP [18] or a transformer-based text encoder converts the prompt into a sequence of contextual

embeddings:

$$\mathbf{T} = E_{\text{text}}(y) = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_L], \tag{3.23}$$

where $\mathbf{t}_i \in \mathbb{R}^d$ represents the i-th token embedding in the dimension of d, E_{text} is the text encoder, and L is the sequence length. These embeddings encapsulate both semantic and syntactic information and serve as conditioning signals for the diffusion U-Net.

Cross-Attention Mechanism

During denoising, the U-Net operates on the noisy latent representation $\mathbf{z}_t$ and generates a cleaner version $\mathbf{z}_{t-1}$ conditioned on $\mathbf{z}_t$, the time step t and the text prompt y. To incorporate textual information, cross-attention layers [59] are inserted in the U-Net's residual block after the convolutional layers [56]. In this way, the cross-attention layers enable the latent visual features to attend to the text embeddings. A cross-attention layer works by computing how strongly each latent image feature should align with each token in the text embedding, where the latent image features act as queries (the part being updated), and the text embeddings act as keys and values. The attention weights determine how much semantic information should be incorporated from each text token, and therefore, every region of the image becomes aware of the corresponding part of the prompt. For example, a canola flower in the image will attend words such as "yellow petal" in the prompt, while it will ignore irrelevant text.

The attention mechanism is denoted as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{3.24}$$

where Q, K, and V denote the query, key, and value matrices, respectively.

In the context of text-conditioning:

$$Q = W_Q\,\mathbf{h}_{\text{img}}, \tag{3.25}$$

$$K = W_K\,\mathbf{T}, \tag{3.26}$$

$$V = W_V\,\mathbf{T}, \tag{3.27}$$

where $\mathbf{h}_{\text{img}}$ are (flattened) intermediate visual features of the U-Net, and W_Q, W_K, W_V are learnable projection matrices. Thus, image features act as queries that attend over text embeddings (the keys and values), effectively transferring semantic context from language to vision. The attended output $\mathbf{h}'_{\text{img}}$ is then computed as:

$$\mathbf{h}'_{\text{img}} = \text{Attention}(W_Q\mathbf{h}_{\text{img}}, W_K\mathbf{T}, W_V\mathbf{T}). \tag{3.28}$$

Here, the learned attention weights (W_Q, W_K and W_V) quantify how much each image location (query) should attend to each token embedding (key), and the result is an attended representation $\mathbf{H}'_{\text{img}}$ that encodes text-guided information relevant to each spatial region in the image.

Integration into the Denoising Network

The attended feature $\mathbf{H}'_{\text{img}}$ is not used to replace the original image feature directly; rather, it is fused with it via a residual connection [60]:

$$\mathbf{H}_{\text{fused}} = \mathbf{H}_{\text{img}} + \mathbf{H}'_{\text{img}}. \tag{3.29}$$

This ensures that the network maintains spatial structure from the latent image while incorporating global semantic context from the text. These fused features are then propagated through convolutional and normalization layers within the U-Net to predict the denoised latent $\mathbf{z}_{t-1}$.

Training Dynamics and Sampling Behavior

During training, the model learns to minimize the expected denoising error (equation 3.22) under random timesteps t. At each step, the entire sequence of token embeddings $\mathbf{T}$ is available to the cross-attention layer. This means that each spatial feature in the latent representation attends to all token embeddings in the prompt.

During inference, the same mechanism applies: the denoising trajectory starts from Gaussian noise $\mathbf{z}_T \sim \mathcal{N}(0, I)$ and iteratively refines it into a coherent image guided by the text embeddings. Thus, the cross-attention layers act as the semantic interface that translates textual meaning into spatially structured visual features throughout the generation process.

3.3.5 Image-Conditioning in the Denoising Process

Beyond text conditioning, diffusion models can be guided by an input image to perform structure-preserving transformation, enabling image-to-image translation. This technique was formalized in SDEdit [23], which demonstrated that meaningful edits can be achieved by adding controlled noise to an input image and then denoising it with a diffusion model. Stable Diffusion extends this principle to the latent space, enabling efficient image-conditioned generation [2] that preserves the semantic structure

of the input image while adapting it to the domain learned by the diffusion model, optionally guided by a text prompt.

Given an input (guide) image $\mathbf{x}$, the image-conditioning process begins by encoding it into the latent space through the VAE encoder $E(\cdot)$:

$$\mathbf{z}_0 = E(\mathbf{x}) \in \mathbb{R}^{C\times H\times W}. \tag{3.30}$$

Rather than starting the reverse diffusion process from pure noise, a noisy latent representation is obtained by sampling at an intermediate timestep $t \in \{0, \ldots, T\}$:

$$\mathbf{z}_t = \sqrt{\alpha_t}\,\mathbf{z}_0 + \sqrt{1-\alpha_t}\,\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \tag{3.31}$$

where the noise level is controlled by the timestep t (or equivalently, the "strength" parameter in Stable Diffusion's `Img2Img` pipeline). A small t preserves more structure from $\mathbf{x}$, while a large t allows more freedom for semantic changes.

The reverse denoising process then reconstructs a new image from $\mathbf{z}_t$, optionally guided by a text prompt $\mathbf{c}$ through cross-attention:

$$\mathbf{z}_{t-1} = \mathbf{z}_t - \sqrt{1-\alpha_t}\,\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}), \tag{3.32}$$

iterating until $\mathbf{z}_0$ is reached. ϵ_θ is the nueral network that predicts the added noise $\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c})$. The final output image is obtained by decoding the latent representation with the VAE decoder $D(\cdot)$:

$$\hat{\mathbf{x}} = D(\mathbf{z}_0). \tag{3.33}$$

This mechanism provides a controllable balance between structural preservation and generative freedom. When t is small, the model performs subtle edits that retain fine-grained details; when t is large, the model can produce substantially altered images while maintaining semantic consistency. This makes image-conditioned diffusion

suitable for tasks such as indoor-to-outdoor plant translation, which is explored in Chapter 5.

3.4 DreamBooth for Personalized Image Translation

DreamBooth [19] is a fine-tuning technique designed to personalize large text-to-image diffusion models with only a few example images. By injecting subject-specific knowledge into the generative process, DreamBooth enables the translation of new inputs into semantically consistent outputs that preserve the identity of the target subject while adapting to different contexts. This section reviews its training objective, model architecture, and loss formulations.

3.4.1 Translation Objective and Model Architecture

The goal of DreamBooth is to adapt a pre-trained latent diffusion model (*e.g.*, Stable Diffusion) to generate personalized instances of a given subject. Given a small set of example images $\{\mathbf{x}_i\}_{i=1}^{N}$ and an associated rare identifier token s^* (*e.g.*, "sks canola plant"), the objective is to fine-tune the model such that prompts containing s^* generate faithful depictions of the subject.

The fine-tuning process retains the original LDM architecture [2], consisting of: (1) a VAE for latent encoding/decoding, (2) a U-Net denoising network with cross-attention for text conditioning, and (3) a text encoder (e.g., CLIP [18]) for embedding the prompts.

During training, DreamBooth modifies the weights of the U-Net (and optionally the text encoder) to align the rare token s^* with the subject's identity while preserving the generative prior of the original model.

3.4.2 Subject Reconstruction Loss

To ensure the fine-tuned model reproduces the subject accurately, DreamBooth employs a reconstruction loss that minimizes the difference between the denoised latent $\hat{\mathbf{z}}_0$ and the true latent $\mathbf{z}_0$ of the subject image:

$$\mathcal{L}_{\text{rec}} = \mathbb{E}_{t,\mathbf{z}_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t, y_{s^*})\|^2\right], \tag{3.34}$$

where ϵ_θ is the noise prediction network conditioned on the subject-specific prompt y_{s^*}. This encourages the model to associate the rare token s^* with the visual features of the target subject.

3.4.3 Class Prior Preservation Loss

A challenge in subject-driven fine-tuning is overfitting, where the model forgets the general class (*e.g.*, "canola plant") while over-specializing to the specific instance (*e.g.*, "sks canola plant"). To mitigate this, DreamBooth introduces a class prior preservation loss by generating synthetic class images using the original model with prompts like "a photo of a canola plant". Formally, the objective is:

$$\mathcal{L}_{\text{class}} = \mathbb{E}_{t,\mathbf{z}_0^{\text{cls}},\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t^{\text{cls}}, t, y_{\text{cls}})\|^2\right], \tag{3.35}$$

where $\mathbf{z}_0^{\text{cls}}$ are latents of class images and y_{cls} is the class prompt. This ensures that the model retains the ability to generate diverse instances of the general class while

incorporating subject-specific details.

3.4.4 Total Loss Function

The overall DreamBooth objective combines the subject reconstruction loss with the class prior preservation loss into a single weighted objective. The total training loss can be expressed as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \lambda \mathcal{L}_{\text{class}}, \tag{3.36}$$

where $\mathcal{L}_{\text{rec}}$ denotes the subject reconstruction loss, $\mathcal{L}_{\text{class}}$ is the class prior preservation loss, and λ is a weighting hyperparameter controlling the trade-off between subject fidelity and generalization. A larger λ strengthens class-level preservation at the cost of subject specificity, while a smaller λ increases subject fidelity but risks overfitting. This weighted combination allows DreamBooth to balance accurate personalization of the subject with the generative model's expressive capacity.

Chapter Summary

In this chapter, we reviewed the theoretical underpinnings of the generative models applied throughout this thesis. We first examined VAEs, which provide a probabilistic framework for latent representation learning. We then introduced the foundations of diffusion models, emphasizing the roles of convolutional neural networks and the U-Net architecture in the iterative denoising process. This is followed by describing how the combination of VAE and diffusion mechanisms gives rise to LDMs. Finally, we discussed DreamBooth as an extension of diffusion models for personalized,

subject-specific fine-tuning and image translation. Together, these components establish the theoretical groundwork for the methodologies developed in the following chapters. Chapters 4 and 5 build upon these concepts by detailing the experimental design, pipeline architectures, and fine-tuning strategies for plant image generation and indoor-to-outdoor translation tasks.

Chapter 4

Image Generation

4.1 Introduction

In this chapter, we describe the methodology and results of fine-tuning Stable Diffusion-v1.4 (SD-v1.4) for plant image generation. Our focus is on adapting the model to the agricultural domain and systematically evaluating the utility of the generated images in downstream classification tasks. This provides insight into the potential of generative models as data augmentation tools for advancing agricultural machine learning.

A high-level flowchart of the image generation pipeline is illustrated in Figure 4.1. The pipeline consists of three main components: **caption generation**, **model fine-tuning**, and **text-to-image synthesis**, as highlighted in Figure 4.1. In the first stage (1), image captions are generated automatically using a multimodal large language model ChatGPT-4o [61], which describes visual and contextual attributes of plant images, such as species, growth stage, and environmental conditions. These

captions form paired text–image datasets used to condition the diffusion model. In the second stage (2), a pretrained SD-v1.4 model is fine-tuned using the curated plant image–caption pairs. Only the U-Net denoiser is updated during training, while the VAE and text encoder remain frozen. In the final image inference stage (3), user-defined prompts are fed into the fine-tuned model to generate new images.

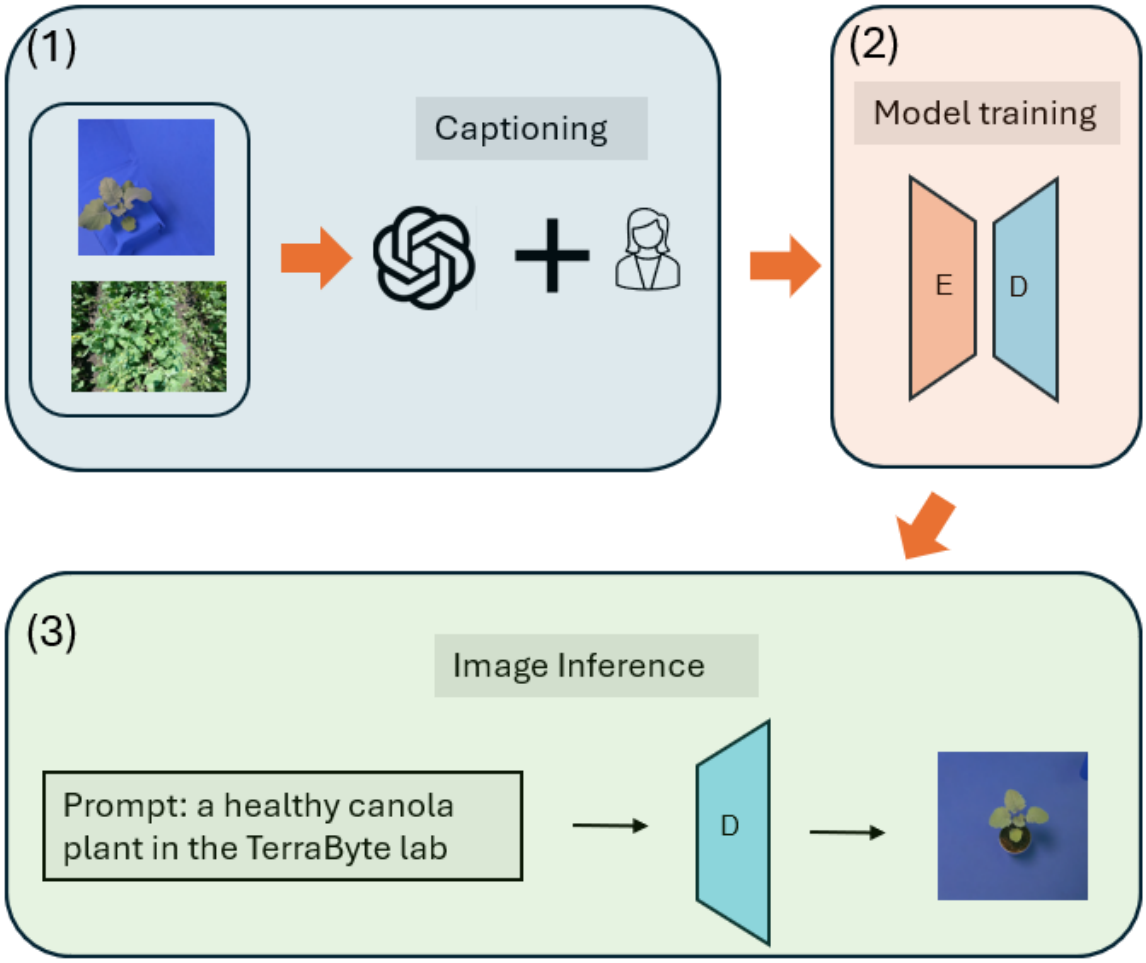

Figure 4.1: Overview of the image generation pipeline. The workflow consists of three components: (1) caption generation using a multimodal LLM, (2) fine-tuning of Stable Diffusion on plant image–caption pairs, and (3) text-conditioned image synthesis for generating realistic crop images.

4.2 Materials and Methods

4.2.1 Datasets

Indoor Plant Images

Our indoor dataset was generated using the EAGL-I system developed by Beck *et al.* [62], which captures high-resolution images of crop plants such as canola and soybean. Representative examples are shown in Figure 4.2. The images were collected in a controlled laboratory environment with standardized lighting and a uniform blue background to reduce visual variability. Each image is accompanied by detailed metadata describing the plant species, growth stage, and camera parameters such as angle and distance. For this study, we focus specifically on canola and soybean as representative crop species.

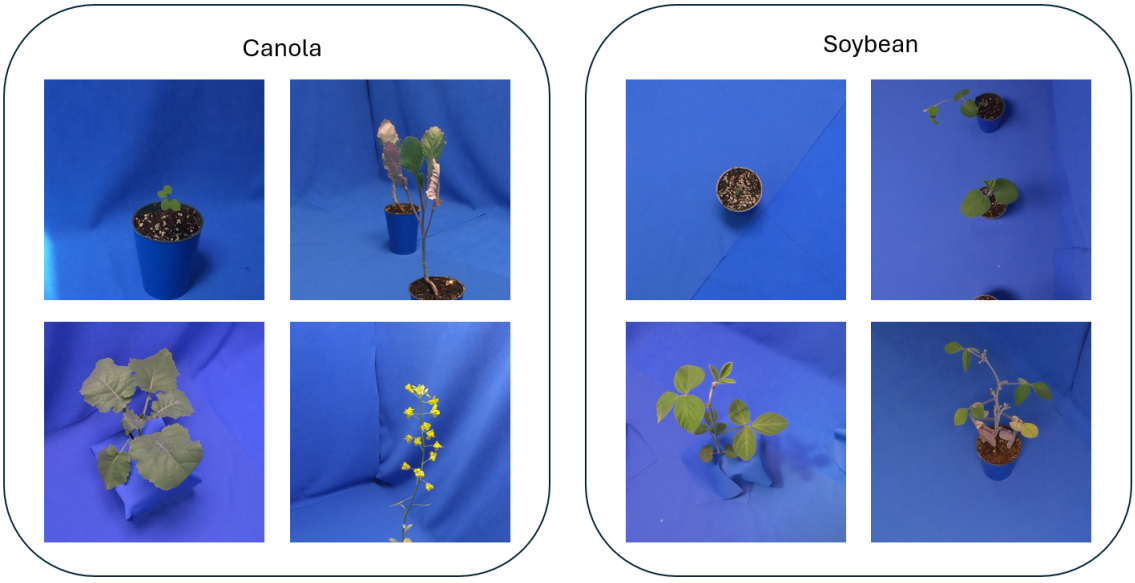

Figure 4.2: Examples of indoor canola and soybean images in different growth stages or conditions.

Outdoor Field Plant Images

Outdoor images were obtained from the TerraByte research group [63], featuring crop scenes captured at EMILI's Innovation Farm in Grosse Isle, Manitoba, Canada. These images reflect the complexity of real-world field conditions, including variability in lighting, background, occlusion, and weather. Data collection was conducted using GoPro cameras connected on a tracker boom. Some images exhibit noise such as motion blur, background machinery, harsh shadows, or empty field, requiring preprocessing to filter out low-quality samples (as illustrated in Figure 4.3). The dataset covers diverse growth stages and environmental conditions for both canola and soybean.

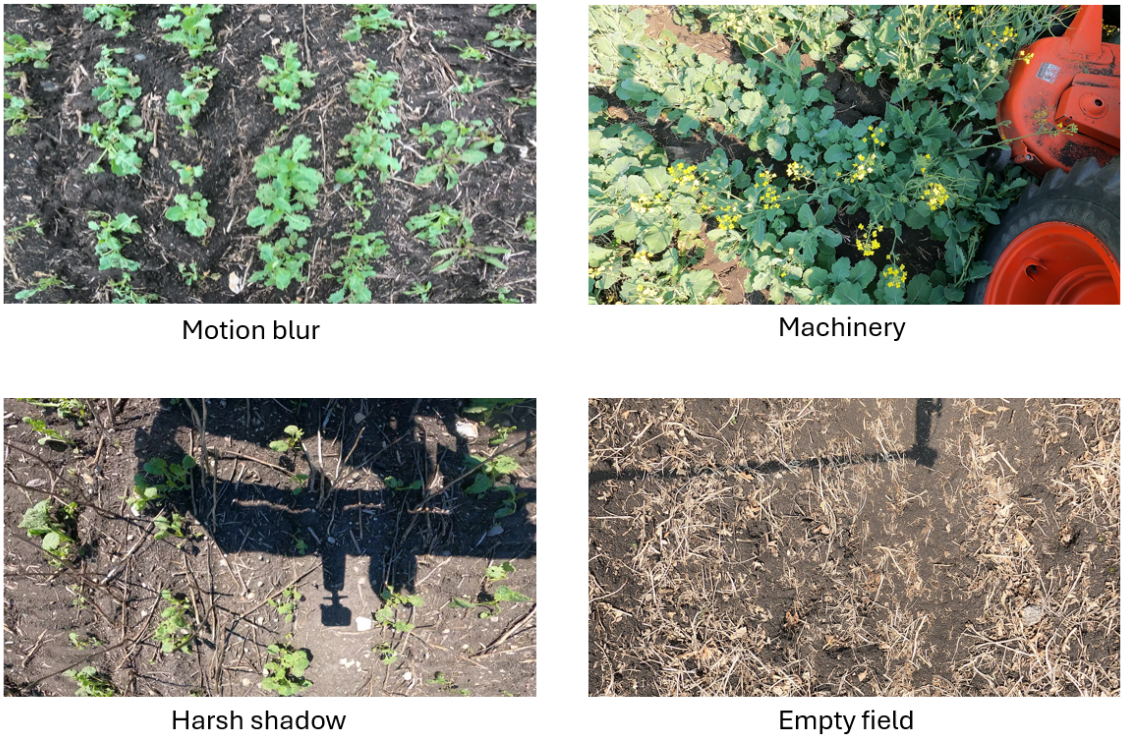

Figure 4.3: Examples of low-quality outdoor canola images across different growth stages and conditions, illustrating representative challenges such as motion blur, background machinery, harsh shadows, and empty fields.

Caption Generation using ChatGPT-4o

To enable text-conditioned training of Stable Diffusion, image-caption pairs were required. Both indoor and outdoor canola plant images were captioned using the multimodal large language model ChatGPT-4o [61], leveraging its image-to-text capability. Several captioning models were initially considered, including BLIP-2 [64], LLaVA [65], and ChatGPT-4o, with the latter chosen for its superior accuracy in describing plant features. The accuracy of the captioning was determined subjectively by human inspection using a representative sample of images across all models.

Prompt engineering was applied to guide caption generation, with emphasis on growth stage, species, and experimental conditions (*e.g.*, whether the image was captured indoors or in the field). An example prompt, input image, and resulting caption are shown in Figure 4.4. Descriptive captions were generated to highlight visible traits, such as: *"Young canola seedling, showing mild yellowing of leaves consistent with nutrient deficiency."* These captions provide domain-specific conditioning signals that guide the denoising process during fine-tuning, improving controllability on generation of plant phenotypes.

Preprocessing: Resizing, Filtering, and Data Splits

The preprocessing pipeline consisted of image quality control, resizing, and dataset partitioning. Low-quality samples (*e.g.*, images with blur or harsh shadows) were removed to ensure consistency. The curated dataset was then randomly divided into training and test sets using a 90/10 ratio. This dataset served as the foundation for fine-tuning the text-to-image generation model as well as for subsequent indoor-to-

```
prompt = f"""
    You are an expert botanist and image analyst. This is a photo of canola plant
taken indoor with a blue background. Describe the photo in detail.

    Key Aspects to Cover:
    1. Plant Species - {species}.
    2. Image Quality - Note if the image is blurry.
    3. Plant Count & Obstructions - How many plants are visible? Any blockages?
    4. Viewpoint - Choose from top-down, oblique, or profile.
    5. Growth Stage - Estimate the plant's age based on visual cues.
    6. Health State - Identify any issues (e.g., pests, yellowing, curling,
uprightness, viral symptoms).

    Carefully integrate the provided information while analyzing the image, use
less than 60 words in total.
    """
```

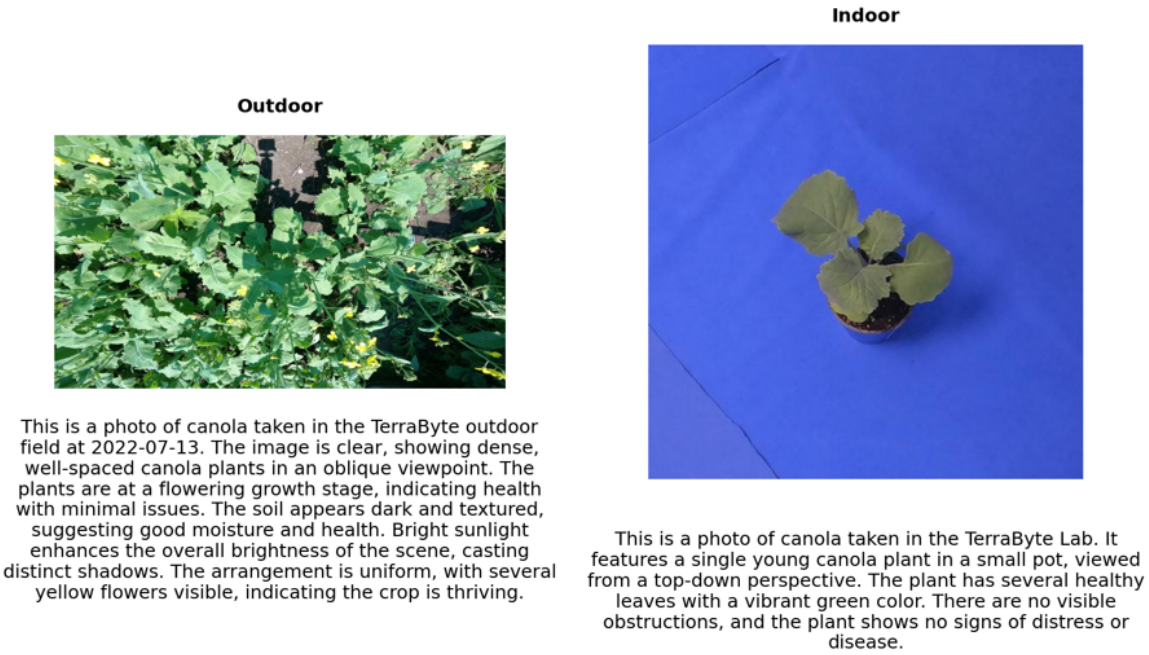

Figure 4.4: Example of the prompt template and caption generation process using ChatGPT-4o. Shown are sample indoor and outdoor canola plant images alongside their automatically generated captions. The prompt was designed to highlight plant species, growth stage, and environmental context, ensuring informative conditioning text for fine-tuning Stable Diffusion model.

outdoor translation experiments. The training set contained a total of around 8,000 canola plant images. Before training, all images were resized to a fixed resolution of 512×512 pixels.

4.2.2 Diffusion Model Fine-Tuning for Text-to-Image Generation

To generate realistic plant images from textual prompts, we fine-tuned a pre-trained LDM on a domain-specific dataset of plant images paired with textual descriptions. Our implementation is based on the SD-v1.4 architecture [2]. By adapting the model to agricultural imagery, the framework is able to capture visual and semantic characteristics unique to canola plants, including morphology, color variation, and environmental context. The fine-tuned model is conditioned on text prompts that describe plant attributes, stress conditions, or growth stages (*e.g.*, *"A photo of canola plant with yellowing leaves under nutrient stress."*).

Base Model Architecture

We begin with SD-v1.4, originally trained on the large-scale LAION-5B dataset [66]. The architecture consists of three main components: (1) a VAE [49], which encodes images into a compact latent space and reconstructs them back to pixel space; (2) a U-Net-based denoising model [56], which performs iterative noise removal during the reverse diffusion process; and (3) a CLIP-based text encoder [18], which converts input prompts into conditioning vectors.

Fine-Tuning Strategy

For domain adaptation, we fine-tune only the U-Net backbone of the latent diffusion model, while keeping the autoencoder and text encoder frozen. This strategy reduces computational cost and prevents catastrophic forgetting of the general vi-

sual and linguistic knowledge encoded in the pretrained components. Training is performed using paired image–caption samples from our curated canola dataset. We adopt the standard DDPM objective, where the network learns to predict the Gaussian noise added to latent image representations. Conditioning is provided via text embeddings generated by the CLIP encoder. The model architecture details are mentioned in Chapter 3.

Sampling

To generate a new image, one samples from a standard Gaussian prior $z_T \sim \mathcal{N}(0, \mathbf{I})$, and iteratively denoises using the learned reverse diffusion process [2]. Then the final latent z_0 is decoded into pixel space using the VAE decoder:

$$\hat{x} \approx \mathrm{Dec}_{\mathrm{VAE}}(z_0). \tag{4.1}$$

An illustration of details of the model training and inferences is shown in Figure 4.5.

Fine-Tuning Setup

The fine-tuning was performed on three NVIDIA A100 GPUs at the University of Manitoba. Training was conducted on the curated training set for a total of 3,000 steps. We used a batch size of 20 with gradient accumulation over 4 steps to effectively increase the batch size without exceeding GPU memory limits. The learning rate was fixed at 1×10^{-5}, and the input resolution was standardized to 512×512 pixels. Early stopping was guided by qualitative inspection: at every 200 training steps, the

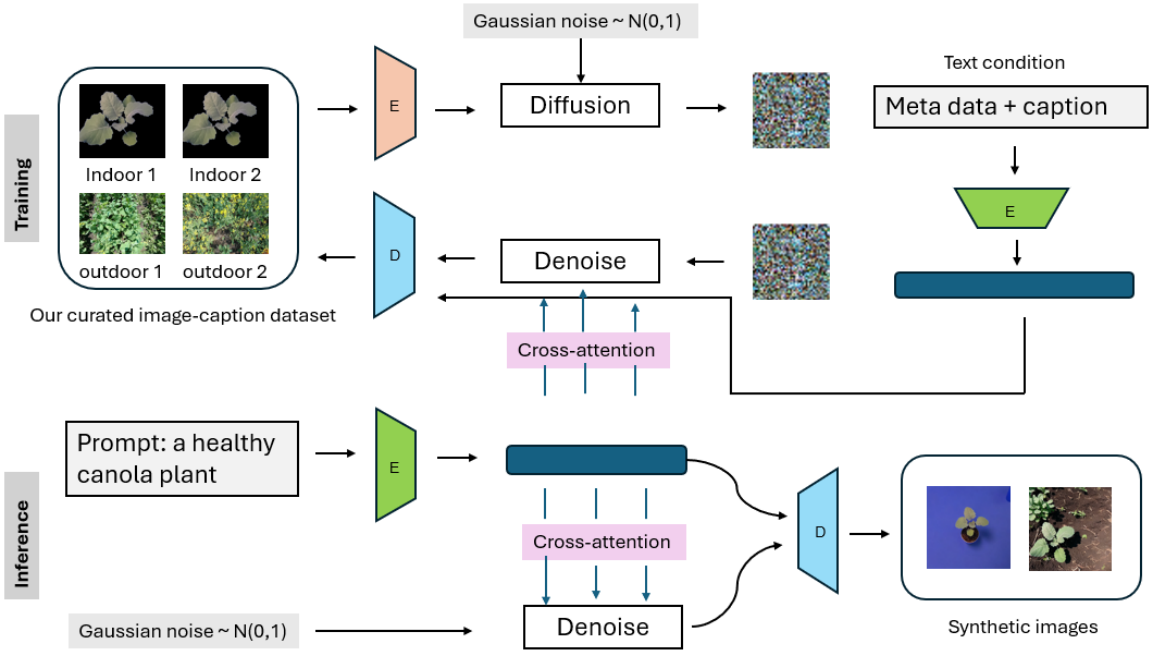

Figure 4.5: Fine-tuning and inference process in our text-to-image pipeline. The top row illustrates model fine-tuning using our curated dataset of canola plant images and corresponding captions. The bottom row depicts the inference phase, where the fine-tuned model generates realistic plant images conditioned on user-provided text prompts.

model generated samples from fixed prompts from the validation set, which were then evaluated for diversity, realism, and consistency with the prompt descriptions.

Sampling Parameters

During inference phase, text prompts from the test dataset were used to condition the image generation process. To optimize the quality of generated images, we conducted a grid search over key sampling parameters:

- **Guidance Scale:** Controls the trade-off between fidelity to the text prompt and visual diversity. Higher values enforce stronger alignment with the prompt but can reduce diversity, while lower values allow more variety at the expense

of semantic accuracy.

- **Number of Inference Steps:** Refers to the number of denoising iterations during sampling. More steps generally improve image quality but increase computational cost.

Through visual inspection of generated samples, we found that a guidance scale of 1.5 and 70 inference steps provided the best balance between prompt fidelity, image realism, and computational efficiency.

4.2.3 Evaluation Metrics, Baseline Models Comparison, and Results

Evaluation Metrics

To assess the quality of generated images, we employ two widely used metrics in generative modeling: the Fréchet inception Distance (FID) [67] and the Inception Score (IS) [68]. FID measures the similarity between the distribution of generated images and real images by comparing the mean and covariance of features extracted from a pre-trained inception network. A lower FID score indicates that the generated images are closer in distribution to real images, reflecting higher fidelity and diversity.

The IS score, on the other hand, evaluates both the diversity and quality of generated samples by computing the KL divergence between the conditional label distribution and the marginal distribution predicted by the inception network[68]. A higher IS score indicates that the generated images are both meaningful (high confidence predictions) and diverse across classes. These metrics provide a complementary

evaluation of realism and variety in synthetic imagery.

Baseline Models

For benchmarking, we compare our fine-tuned SD-v1.4 model against three state-of-the-art generative models.

- **Imagen** is a diffusion-based text-to-image model that achieves photorealistic quality through large-scale training and hierarchical generation. Although it sets a strong baseline for natural image synthesis, Imagen has not been explicitly trained in agricultural domains [17].

- **DALL·E** is a transformer-based generative model that integrates discrete variational autoencoders with autoregressive modeling. It is capable of producing diverse images from natural language descriptions but may lack fine control over structural details [16].

- **GigaGAN** represents a recent advancement in GAN-based image generation. It is designed for large-scale high-resolution synthesis. However, GAN-based models are often less stable than diffusion-based approaches and may produce artifacts in domains with high structural variability [69].

Results

Figure 4.6 presents examples of generated canola plant images under both indoor and outdoor environmental settings. The indoor samples exhibit high visual fidelity and diversity, capturing fine-grained details such as leaf texture, lighting variations and growth stages. Meanwhile, the outdoor samples demonstrate strong realism

and controllability, with image semantics modulated through text prompts. The generation process can capture distinct growth stages, such as those corresponding to June 2, June 21, July 4, and July 22, illustrating the model's ability to synthesize temporally consistent plant development under varying field conditions.

Figure 4.7 and Table 4.1 summarize the performance comparison across indoor and outdoor canola plant datasets. For the IS score, our model achieves the highest score of 3.29 on indoor images, outperforming Imagen (3.04), DALL·E (2.89), and GigaGAN (2.44). On the outdoor dataset, Imagen slightly outperforms our model (2.72 vs. 2.60), while DALL·E (2.21) and GigaGAN (2.06) achieve significantly lower scores. The overall trend indicates that indoor images yield higher IS values than outdoor images, likely due to the reduced complexity and controlled conditions of laboratory settings.

For FID, our model consistently achieves the best results in both domains. On the indoor dataset, we obtain a score of 83.4, compared to Imagen (118.2), DALL·E (103.9), and GigaGAN (98.4). On the outdoor dataset, our approach again performs best with 127.6, followed by GigaGAN (133.1), DALL·E (129.2), and Imagen (149.1). Although all models show higher FID values for outdoor images—indicating increased difficulty in modeling field variability—our fine-tuned diffusion model demonstrates clear advantages in both fidelity and diversity.

4.3 Downstream Machine Learning Task

To further validate the effectiveness of our image generation model, we designed a downstream machine learning task in which synthetic images are used for data

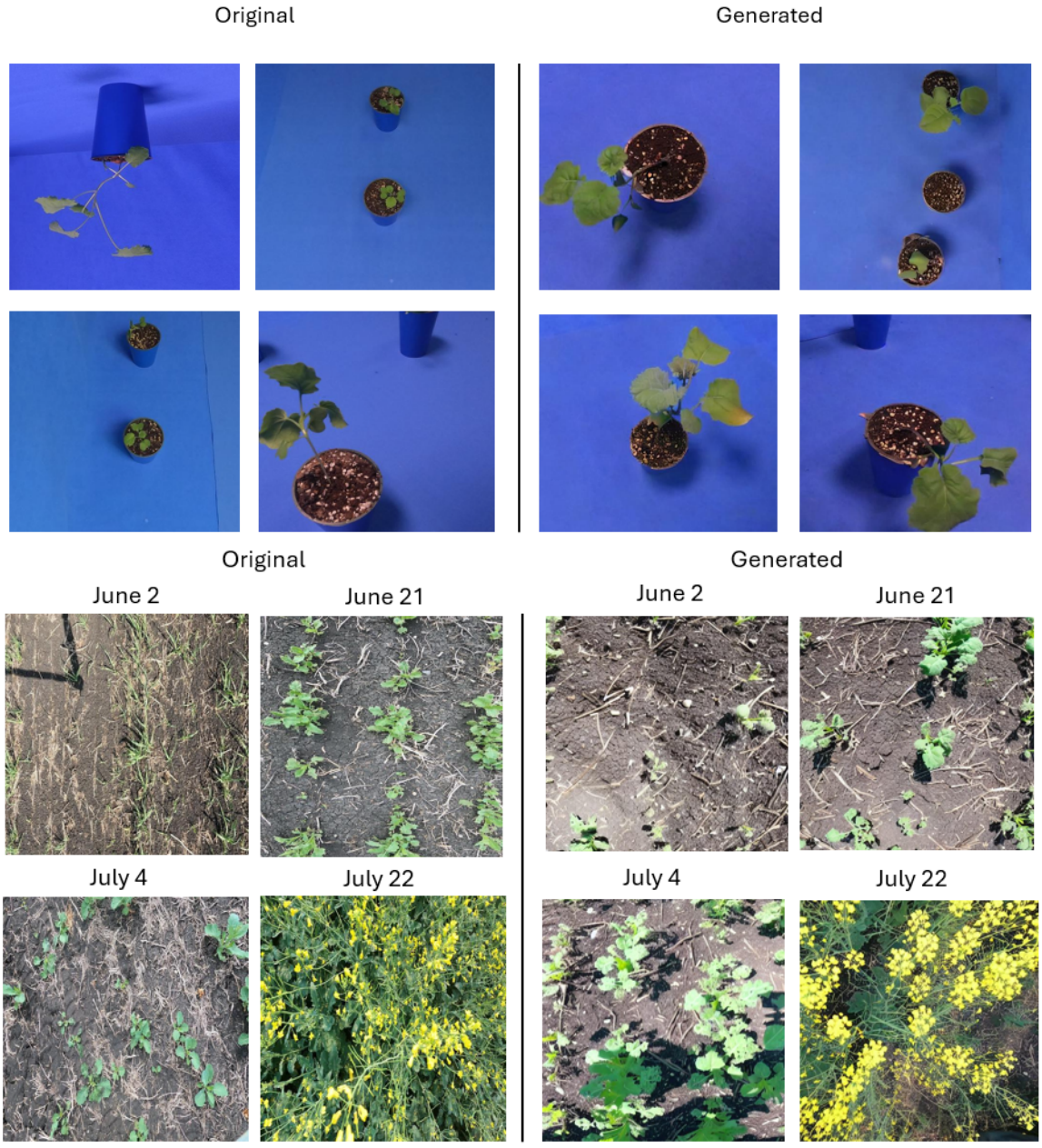

Figure 4.6: Examples of generated indoor and outdoor canola plant images. The results highlight the flexibility of the fine-tuned Stable Diffusion model in capturing plant morphology, color variation, growth stage and environmental context. The generated samples also demonstrate controllability through prompt design.

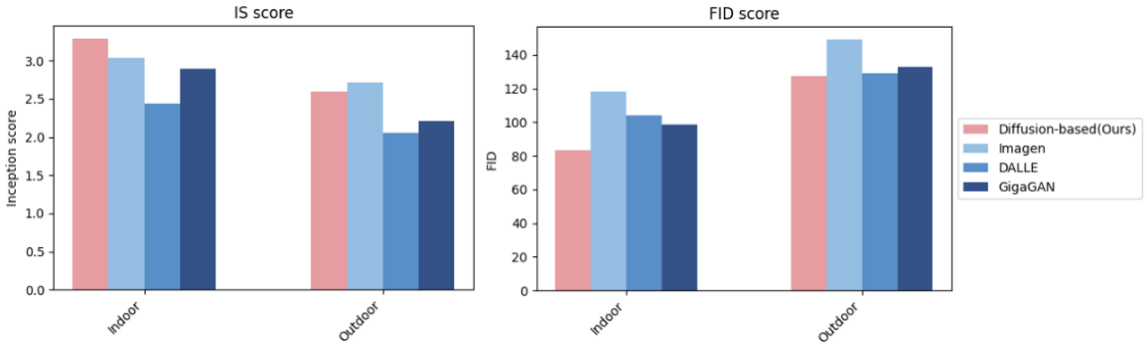

Figure 4.7: Evaluation results of fine-tuned Stable Diffusion compared with baseline models. Performance is reported for both indoor and outdoor canola datasets using Inception Score (IS) and Fréchet Inception Distance (FID). Higher IS and lower FID indicate better generative performance.

Table 4.1: Comparison of IS and FID across models for indoor and outdoor canola plant datasets. Best results are highlighted in bold.

Model	IS ↑		FID ↓	
	Indoor	Outdoor	Indoor	Outdoor
Imagen	3.04	**2.72**	118.20	149.10
GigaGAN	2.44	2.06	98.44	133.06
DALL·E	2.89	2.21	103.89	129.21
Ours	**3.29**	2.60	**83.40**	**127.60**

augmentation. The objective was to assess whether the inclusion of generated images improves the performance of phenotype classification models compared to training solely on real data.

4.3.1 Benchmark Datasets for Phenotype Classification

We employed two benchmark datasets commonly used in plant disease recognition:

- **PlantVillage dataset** [70]. This dataset consists of leaf images from three

species—tomato, potato, and bell pepper—covering a wide range of healthy and diseased phenotypes. Tomato leaves include ten phenotypes, potato includes three, and bell pepper includes two. The distribution of training and test samples for each phenotype is shown in Table 4.2.

- **CropDisease dataset** [71]. This dataset includes leaf images from maize, cashew, and cassava, with seven, five, and five phenotypes respectively. Details of the training and test set sizes are provided in Table 4.3.

For both datasets, we adopted a training/test split of 70/30 to ensure balanced evaluation across phenotypes. Classification models were trained independently for each species, enabling a direct assessment of the impact of synthetic image augmentation on diverse crops and phenotypic classes.

4.3.2 Experiment Setup

The experimental design for the downstream machine learning tasks consisted of four main stages:

1. **Stable Diffusion Fine-Tuning.** A pre-trained SD-v1.4 model was fine-tuned using the PlantVillage and CropDisease datasets. This process followed the same text-to-image generation pipeline described in Section 4.1, with the main difference being the choice of captioning model. Here, image captions were generated using `llava-1.5-13b-hf` (Llava-1.5)[65], an open-source vision–language model. LLaVA-1.5 was selected because the captioning requirements for these datasets are relatively simple, and the model provides strong captioning accuracy while remaining computationally efficient and free from token-based usage

Table 4.2: Number of images in the training and test sets for tomato, potato, and bell pepper phenotypes in the PlantVillage dataset.

Tomato phenotypes	**Train**	**Test**	**Total**
Healthy	1012	434	1446
Tomato mosaic virus	259	111	370
Leaf mold	630	270	900
Early blight	689	296	985
Target spot	1031	442	1473
Spotted spider mite	1180	506	1686
Septoria leaf spot	1200	515	1715
Late blight	1377	590	1967
Bacterial spot	1512	648	2160
Yellow leaf curl virus	2230	957	3187
Potato phenotypes			
Healthy	117	50	167
Early blight	648	277	925
Late blight	739	317	1056
Bell pepper phenotypes			
Healthy	1056	453	1509
Bacterial spot	761	326	1087

costs. The resulting image–caption pairs were then used to fine-tune pre-trained SD-v1.4 model with the same hyperparameter settings described earlier in Section 4.1.

Table 4.3: Number of images in the training and test sets for maize, cashew, and cassava phenotypes in the CropDisease dataset.

Maize phenotypes	**Train**	**Test**	**Total**
Healthy	121	52	173
Fall armyworm	196	84	280
Grasshopper	484	207	691
Leaf beetle	609	261	870
Steak virus	645	277	922
Leaf blight	711	305	1016
Leaf spot	860	369	1229
Cashew phenotypes			
Healthy	988	424	1412
Gummosis	280	120	400
Leaf miner	968	415	1383
Red rust	1194	513	1707
Anthracnose	1226	526	1752
Cassava phenotypes			
Healthy	760	326	1086
Green mite	875	375	1250
Mosaic	885	380	1265
Brown spot	1032	442	1474
Bacterial blight	1697	728	2425

2. **Synthetic Image Generation.** After fine-tuning, synthetic images were generated from the captions corresponding to the test set. To ensure fair evaluation, the original test set was further divided into a 60/40 split. Synthetic images were generated from captions in the 60% portion, while the final evaluation of the classification models was performed exclusively on the remaining 40%, which contained only real images. For each caption, multiple synthetic images were generated to increase diversity.

3. **Phenotype Classification Model.** For each crop species, we trained a classification model using the same backbone architecture, a custom CNN. The network was designed specifically for balancing depth and efficiency to suit small-to-medium-sized agricultural datasets. As summarized in Table 4.4, the model consists of four convolutional blocks, each followed by batch normalization, ReLU activation, and downsampling. The first three convolutional layers use 3×3 kernels with padding, followed by 2×2 max pooling layers that progressively reduce the resolution from 224×224 to 28×28. The final block expands the number of filters to 256 and applies an adaptive average pooling layer. Extracted features are flattened and passed through a fully connected classifier with dropout (0.5) for regularization, followed by a final linear layer mapping to the number of target classes.

4. **Model Training and Data Augmentation Design.** To assess the contribution of synthetic data, we evaluated the classification model under different levels of data augmentation. Specifically, we tested synthetic-to-real ratios of {0%, 50%, 100%, 200%, 300%, 400%}. At 0%, the model was trained only on

real images. At 50%, synthetic images were added equal to half the size of the real dataset. For higher ratios, the number of synthetic images was progressively increased while keeping the set of images from the lower ratio fixed. For example, the 100% condition added additional 50% synthetic images relative to the 50% case, and the same rule was applied for 200–400%. All synthetic images were sampled from the generated pool corresponding to the 60% split of the test set, ensuring consistency and fairness across augmentation levels.

Table 4.4: Summary of CNN architecture used as the downstream machine learning task to assess synthetic image quality.

Stage	**Configuration**
Input	$3 \times 224 \times 224$ RGB image
Feature extractor	Conv(32, 3×3) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ MaxPool(2)
	Conv(64, 3×3) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ MaxPool(2)
	Conv(128, 3×3) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ MaxPool(2)
	Conv(256, 3×3) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ AdaptiveAvgPool(1)
Classifier	Flatten $\rightarrow$ Dropout(0.5) $\rightarrow$ Linear(256 $\rightarrow$ num_classes)

4.3.3 Generated Images for Each Phenotype

To qualitatively assess the performance of our fine-tuned SD-v1.4 model, we compared generated images with real samples across different phenotypes. Representative examples are shown in Figure 4.8 for tomato leaves and Figure 4.9 for maize leaves. As illustrated in Figure 4.8, the synthetic tomato leaves capture diverse disease phenotypes with realistic textures, color variations, and lesion patterns that closely resemble

their real counterparts. Similarly, Figure 4.9 demonstrates that the generated maize leaves reflect characteristic traits of different phenotypes with convincing fidelity.

Overall, the generated images are of high visual quality and in many cases are difficult to distinguish from real images. Minor imperfections are occasionally observed, such as unnatural lighting artifacts or slight distortions in pest structures on the leaf surface. Nevertheless, the synthetic images provide strong evidence that the fine-tuned diffusion model can effectively learn phenotype-specific features and generalize across different crop species.

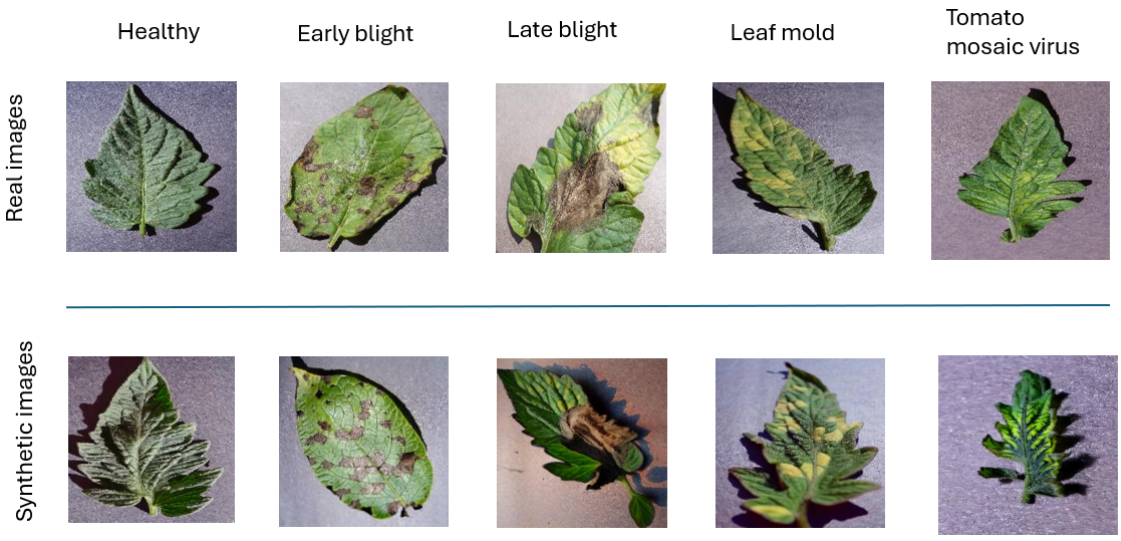

Figure 4.8: Comparison of generated and real tomato leaf images across phenotypes. The fine-tuned Stable Diffusion model produces synthetic leaves with realistic morphology and disease-specific patterns that align closely with real samples.

4.3.4 Evaluation and Results

The performance of the phenotype classification models was evaluated in terms of classification accuracy across varying levels of synthetic data augmentation. The

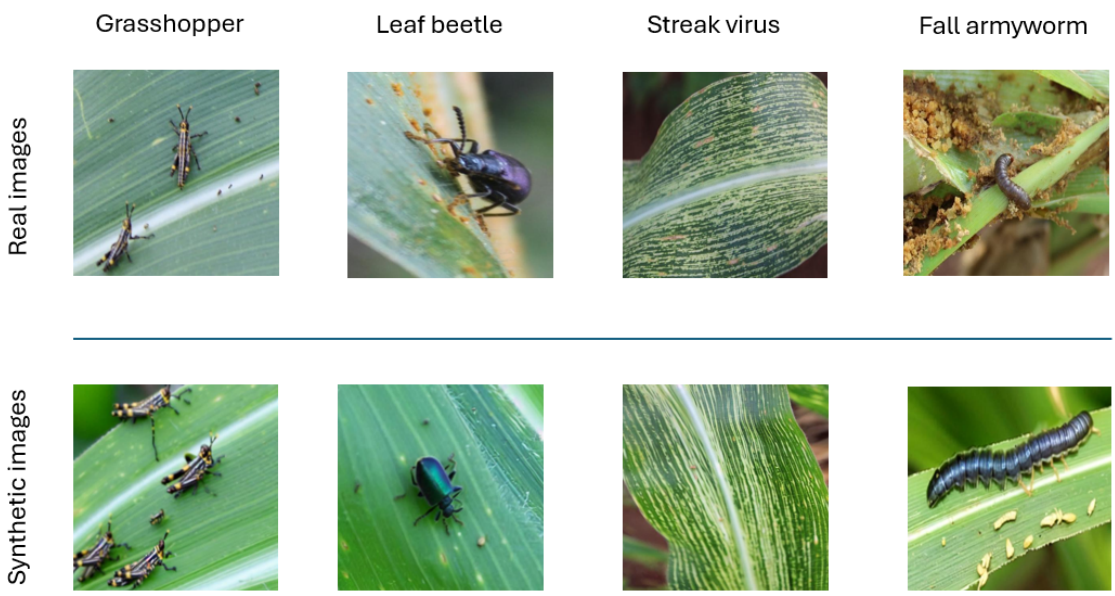

Figure 4.9: Comparison of generated and real maize leaf images across phenotypes. The generated samples demonstrate strong fidelity in reproducing key visual traits of maize phenotypes.

classification accuracy was evaluated exclusively on real images.

PlantVillage dataset. Figure 4.10(a) illustrates the classification accuracy on the PlantVillage dataset across different synthetic-to-real ratios. A general upward trend is observed, with accuracy improving as more synthetic images are incorporated into the training set. The models achieve their highest performance when the synthetic ratio reaches 400% for all three species. Among the crops, tomato consistently exhibits lower accuracy compared to potato and bell pepper. This discrepancy arises because tomato classification is inherently more challenging, involving ten phenotypes, whereas potato and bell pepper classification involves only three and two phenotypes, respectively.

Another key observation is that the performance curves tend to plateau at higher

synthetic ratios, suggesting that model capacity or noise in the dataset imposes an upper bound on achievable accuracy. Detailed classification results are provided in Table 4.5.

CropDisease dataset. A similar trend was observed on the CropDisease dataset (Figure 4.10(b), Table 4.5). Classification accuracy improves steadily as synthetic images are added, although the rate of improvement decreases at higher ratios. Overall, performance on CropDisease is lower than on PlantVillage. This can be attributed to both the lower quality of images in the dataset (see Figure 4.6) and the greater difficulty of the classification task, which requires distinguishing not only between viral and bacterial infections but also between pest-induced damage and abiotic stress symptoms.

Summary. Across both datasets, the addition of synthetic images consistently enhances classification performance, supporting our hypothesis that text-conditioned image generation can serve as an effective form of data augmentation. While diminishing returns are observed at higher ratios, the results demonstrate that synthetic data improves generalization, particularly in settings with limited real data. This finding highlights the practical potential of diffusion-based image generation for enhancing machine learning pipelines in agriculture.

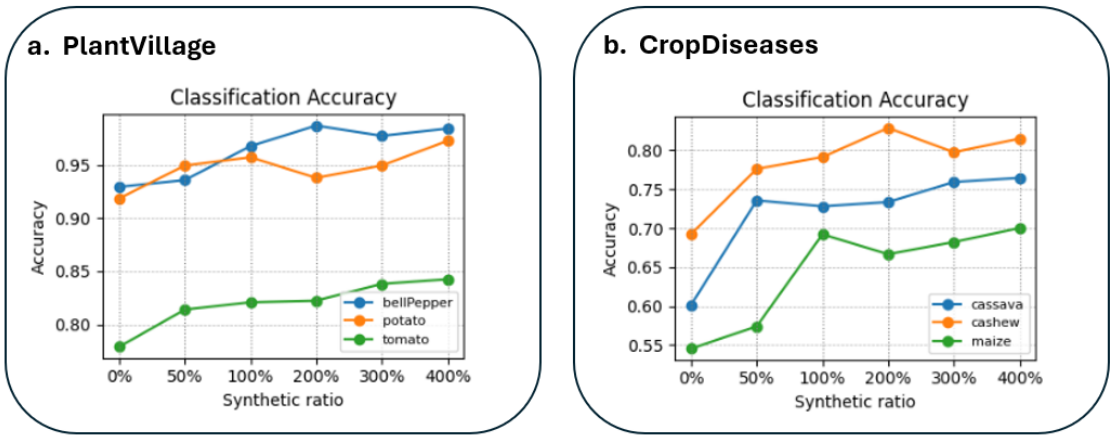

Figure 4.10: Classification accuracy of the custom CNN model with different synthetic-to-real data ratios. (a) Results for the PlantVillage dataset. (b) Results for the CropDisease dataset. Accuracy generally increases with the addition of synthetic images, with diminishing returns at higher ratios.

4.4 Discussion

4.4.1 Strengths of SD Fine-Tuning for Agricultural Data

Our experiments demonstrate that fine-tuning Stable Diffusion on agricultural datasets offers several notable advantages. First, the model effectively captures phenotype-specific features such as disease lesions, discoloration, and pest-induced damage, producing images that are often indistinguishable from real samples. This fidelity enables the use of synthetic data for visualization and training augmentation where collecting field images is expensive or impractical. Second, the text-conditioning capability of the fine-tuned model provides flexibility and controllability, allowing users to generate targeted images of crops under specified conditions (*e.g.*, nutrient deficiency, pest infection, or particular growth stages). Finally, downstream evaluations validate the utility of this approach: incorporating synthetic data con-

Table 4.5: Classification accuracies of the custom CNN model on two benchmark datasets (PlantVillage and CropDisease) under varying synthetic data ratios.

Dataset / Crop	0%	50%	100%	200%	300%	400%
PlantVillage						
Bell pepper	0.9295	0.9359	0.9679	0.9872	0.9776	0.9844
Potato	0.9186	0.9496	0.9574	0.9380	0.9496	0.9729
Tomato	0.7788	0.8139	0.8208	0.8223	0.8381	0.8425
CropDisease						
Cassava	0.6016	0.7358	0.7281	0.7336	0.7592	0.7647
Cashew	0.6925	0.7762	0.7913	0.8287	0.7975	0.8150
Maize	0.5455	0.5742	0.6922	0.6667	0.6823	0.7002

sistently improved classification accuracy across multiple benchmarks, highlighting the potential of generative augmentation to enhance machine learning in agricultural contexts.

4.4.2 Limitations on Domain Bias

Synthetic images from field conditions generally exhibited higher FID values than indoor images, reflecting the greater complexity and variability of outdoor environments. These biases underscore the need for continued curation of training datasets and prompt engineering, especially in outdoor agricultural settings.

4.4.3 Lessons Learned for Image Generation and Downstream Machine Learning Tasks

Several lessons emerged from this part of study. First, high-quality captions are critical for successful fine-tuning, as the alignment between textual descriptions and visual traits directly influences model controllability. Our visual comparisons between different captioning models confirm that careful prompt engineering and captioning model selection significantly impact generative performance.

Second, while synthetic images reliably improved classification accuracy, performance gains tended to plateau at higher augmentation ratios. This suggests that beyond a certain point, model capacity or dataset noise constrains the benefits of synthetic data. Therefore, an optimal balance between real and synthetic samples is necessary to maximize downstream performance.

Finally, our findings emphasize the importance of domain adaptation for generative models in agriculture. While general-purpose models such as Stable Diffusion provide a powerful foundation, tailoring them to agricultural datasets is essential to capture the subtle but critical details of plant phenotypes. Future work should explore related approaches that combine synthetic data with active learning from human feedback, further reducing dependence on costly real-world data collections and annotations.

Chapter 5

Image Translation

5.1 Introduction

Plant image translation from controlled indoor environments to natural outdoor field conditions addresses a critical gap in agricultural machine learning. While large collections of high-resolution indoor plant images are available, outdoor crop images remain comparatively scarce due to the challenges of field data collection, including variable lighting, weather, and field complexity. This imbalance limits the development of robust deep learning models for real-world agricultural applications.

To mitigate this issue, we propose an image translation approach that leverages the generative diffusion model fine-tuned in Chapter 4. Building upon the Stable Diffusion architecture, we introduce a translation component capable of adapting indoor plant imagery to realistic outdoor scenes. This strategy allows us to expand training datasets with synthetic field-like images while preserving plant morphology, thereby improving the generalizability of downstream models for agricultural tasks.

5.2 Materials and Methods

5.2.1 Overview of the Image Translation Task

The image translation component builds upon the fine-tuned diffusion model developed in Chapter 4. Since the model has already been adapted to the agricultural domain using curated image–text pairs, it is capable of associating plant structures with textual semantics. By leveraging this property, we aim to translate indoor plant images into outdoor field scenes by conditioning the denoising process on outdoor-related keywords such as *"TerraByte field," "natural lighting,"* and *"moist soil."* The fine-tuned model's ability to generate realistic outdoor backgrounds, textures, and lighting enables us to create images that preserve the morphology and growth stage of the original indoor plant while embedding it in natural outdoor contexts.

The objective is to retain the structural integrity of the indoor plant (*e.g.*, shape, growth stage, and phenotype) while adapting the surrounding environment to outdoor conditions. This includes environmental variations such as lighting, soil characteristics, and stress conditions (*e.g.*, drought or pest infection). Because the translation process is text-conditioned, it remains controllable by user prompts, allowing customized simulations of diverse field environments.

For this task, we adopt a text-inversion strategy within a DreamBooth fine-tuning framework. As illustrated in Figure 5.1, we introduce a rarely used token (*e.g.*, `sks`) to anchor the semantics of the target plant. The token `sks` was selected because it is not an English word and not associated with existing semantic meaning in the model's vocabulary, minimizing interference with pretrained language concepts. This

token is learned to represent the structural features of the plant in the latent diffusion model. At inference time, prompts combine the rare token with outdoor environment keywords—for example: *"A* ***sks*** *canola plant grows in the TerraByte field, several weeks old and healthy, with moist soil beneath it."* In this way, the indoor plant structure is reconstructed based on the rare token, while the outdoor background is rendered according to the environmental semantics encoded in the prompt. The text-conditioning mechanism provides strong controllability, enabling flexible and user-driven customization of the translation process. The overall workflow can be summarized in four stages:

1. **Input Stage (Indoor Data).** Indoor canola plant images are collected under controlled laboratory conditions. Each image is associated with the rare token `sks`, which is introduced into the text encoder's vocabulary.

2. **Training Stage (DreamBooth Fine-Tuning).** Indoor plant images paired with prompts containing `sks` are used to fine-tune the latent diffusion model. Cross-attention layers in the U-Net bind `sks` to the plant's structural semantics, while a prior preservation pathway simultaneously trains on generic "canola plant" prompts to prevent overfitting and maintain generalization.

3. **Conditioning Stage.** At inference, prompts are constructed using both `sks` (structural semantics) and environment-related tokens such as `TerraByte`.

4. **Translation Stage (Output).** The fine-tuned model generates images in which the plant structure is preserved, but the environment is adapted to outdoor field conditions, resulting in translations of indoor-to-outdoor images.

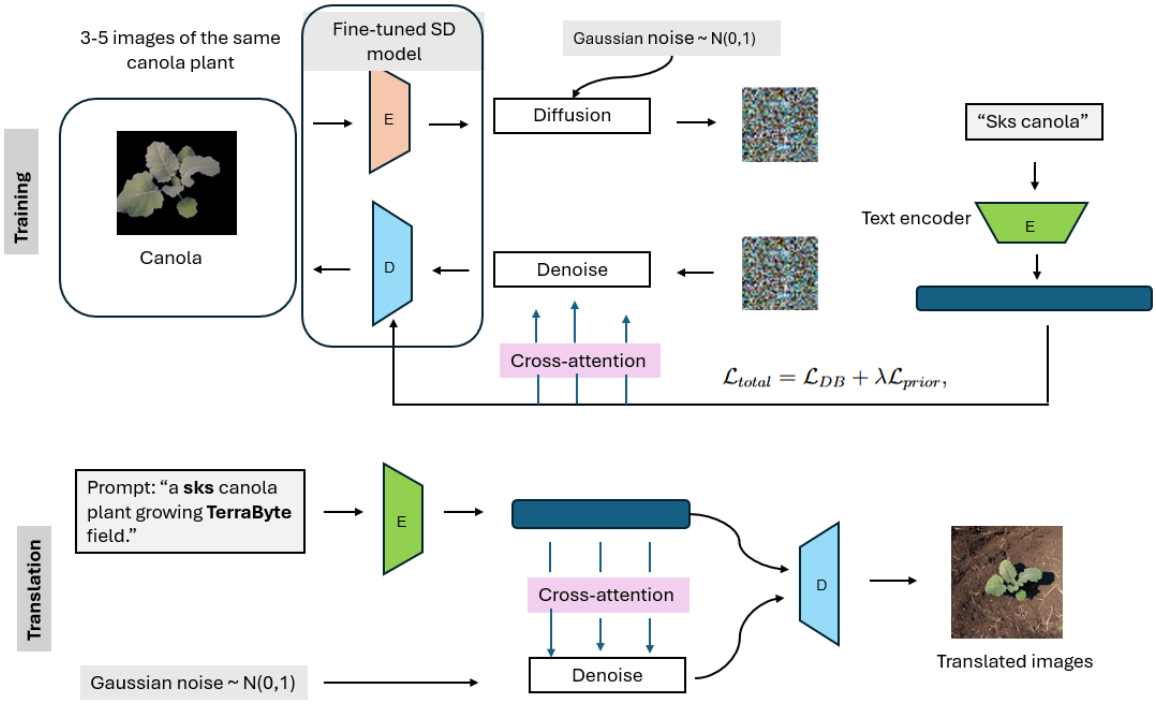

Figure 5.1: Workflow of DreamBooth-based text-conditioned image translation. Indoor canola plant images are paired with the identifier token `sks`, which encodes the structural semantics of the plant. During DreamBooth fine-tuning, the latent diffusion model binds the `sks` token to plant structure via cross-attention, while a prior preservation loss maintains generalization to the broader "canola plant" class. At inference, prompts combine `sks` (structural semantics) with `TerraByte` (outdoor environment semantics). The model then generates translated images where indoor plant structures are preserved but embedded in realistic outdoor field conditions.

5.2.2 Datasets

As introduced in Chapter 4, our indoor dataset is derived from the study by Beck et al. [62], which provides high-resolution images of crop plants such as canola, soybean, and wheat. For the image translation task, we focus specifically on canola as the target crop species.

Each plant image requires preprocessing before being used in the translation pipeline. This step is necessary because the rare token `sks` must be associated

exclusively with the structural semantics of the plant itself, rather than unrelated background elements such as pots or soil. To achieve this, we applied plant segmentation prior to DreamBooth training. Since the indoor images were collected under controlled laboratory conditions with a uniform blue background, a simple color-thresholding approach was sufficient. We implemented segmentation using the open-source Python toolkit PlantCV [72], which is widely used in plant phenotyping research. PlantCV provides modular functions for image analysis, enabling researchers to build customized pipelines for extracting quantitative traits such as leaf area, shape, and color.

For DreamBooth training, it is recommended to use a small set of images (typically 3–5) of the target object captured from different perspectives. To meet this requirement, our preprocessing pipeline consisted of: (1) segmenting and cropping the target plant to remove the background; (2) applying spatial transformations such as flipping to augment the dataset to produce five distinct input images. These preprocessed images were then used to fine-tune the DreamBooth model, with the rare token `sks` serving as the textual representation of the target plant.

5.2.3 DreamBooth for Text-Conditioned Image Translation

DreamBooth is a fine-tuning technique originally developed to personalize large pre-trained text-to-image diffusion models. It enables the model to associate a rare identifier token (e.g., `sks`) with the visual appearance and structural semantics of a specific subject, while preserving the generalization capabilities of the base model.

In our application, the identifier `sks` was introduced to encode the structural fea-

tures of canola plants imaged under controlled indoor conditions. At inference time, this subject-specific token was combined with an environmental descriptor token (e.g., `TerraByte`), which encodes the outdoor field domain. This dual conditioning allows the model to retain the plant's intrinsic structure while adapting the surrounding scene to outdoor environments. As a result, the translation process generates images where indoor plant morphologies are realistically embedded into our target field settings, controlled by text prompts.

5.2.4 DreamBooth Adaptation

The base model for DreamBooth is a LDM. DreamBooth modifies this process by injecting a rare token identifier w_{sks} into the text vocabulary and associating it with images of the target plant. During training, text prompts y are augmented with w_{sks}, so that the cross-attention layers in the U-Net and text encoder bind the plant structure to this token. The optimization loss function becomes:

$$\mathcal{L}_{\text{DB}}(\theta) = \mathbb{E}_{z,\epsilon,t}\left\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y \cup w_{\text{sks}}))\right\|_2^2. \tag{5.1}$$

This enforces the alignment of the `sks` token with the semantic and structural features of the canola plant, allowing it to be recalled and re-contextualized in new prompts. To avoid catastrophic forgetting of the base model's general visual knowledge, DreamBooth training interleaves two strategies, the subject-specific fine-tuning and the class-specific prior preservation, as mentioned in Chapter 3.

5.2.5 Text-Conditioned Translation for Outdoor Adaptation

Once trained, the DreamBooth model associates `sks` with the canola structure. During inference, prompts of the form *"a sks plant growing in the TerraByte field."* guide the model to reconstruct indoor plants within specific outdoor contexts. Here, `TerraByte` acts as a domain keyword capturing the environmental semantics of our target outdoor dataset, while `sks` ensures the translated plant retains its structural integrity. The translation is therefore not a direct pixel-level mapping (as in CycleGAN [10]), but rather a semantic conditioning in the diffusion process, yielding high-fidelity images with controlled structure–environment interaction.

5.2.6 Model Training Setup

The DreamBooth-based fine-tuning for image translation was conducted on three NVIDIA A100 GPUs at the University of Manitoba. The model was trained on the curated indoor canola dataset. Each input plant was cropped, segmented, and augmented into five images. Training was performed for 400 steps, with standard DreamBooth hyperparameters adapted for stable convergence [19]. The VAE and text encoder were frozen during training, while the U-Net backbone and the newly introduced subject-specific token embedding related to `sks` were fine-tuned to align the rare token `sks` with plant structural features.

At inference time, prompts were designed to flexibly customize the translated output according to user requirements. As shown in Figure 5.2, the fine-tuned model preserves the morphology of the input indoor plant while adapting the environment to field conditions. Through prompt engineering, the user can specify details such

as soil characteristics, pest-infected leaves, number and arrangement of plants in the field, or even developmental stage (*e.g.*, translating a plant in a vegetative stage into its flowering stage). This controllability highlights the advantage of text-conditioned translation for generating diverse, user-specific agricultural scenarios.

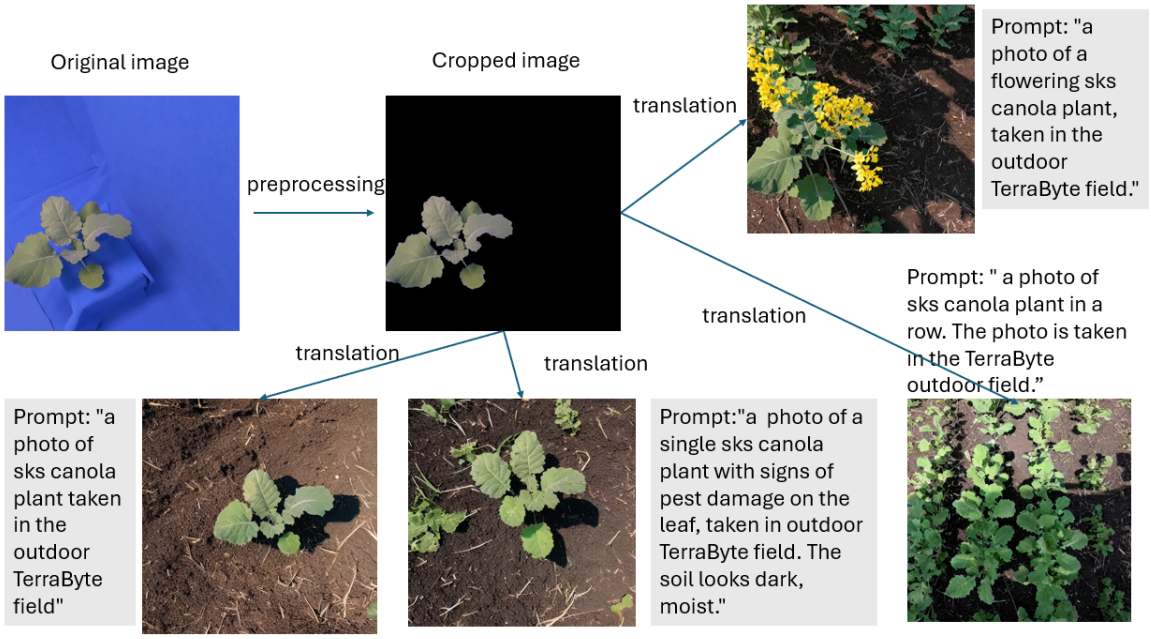

Figure 5.2: Examples of translated canola plant images from indoor to outdoor. Indoor plant images with blue backgrounds were cropped and augmented into five inputs, which were then translated into outdoor field scenes. Prompts enable customization of environmental conditions (*e.g.*, soil type, pest infection, plant density, growth stage), allowing flexible user control over the translation process.

5.2.7 Evaluation Metrics and Results

The quality of the translated images was evaluated using the FID and IS, consistent with the evaluation metrics described in Chapter 4. Figure 5.3 presents the quantitative results. On average, the translated canola images achieved an FID score of 162.1 and an IS score of 2.48. As expected, these scores are lower than those of the

purely generated images reported in Chapter 4. This difference is reasonable because the translation task inherently depends on the generative model to adapt an existing indoor plant structure into a new outdoor environment, which introduces additional sources of variability. Nevertheless, both FID and IS values indicate that the translated images are of reasonably high quality. This observation is further supported by the visual inspection of sample translations shown in Figure 5.2, where plant structures are well preserved and outdoor environments are realistically rendered. While translation scores are lower than those achieved by direct image generation, they remain sufficiently strong to support downstream agricultural tasks.

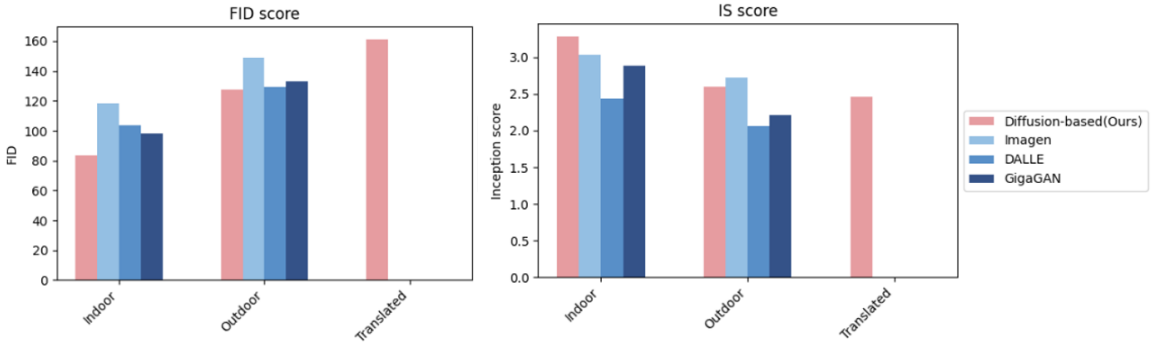

Figure 5.3: Quantitative evaluation of translated canola images. The results are reported in terms of Fréchet Inception Distance (FID) and Inception Score (IS). Lower FID and higher IS values correspond to better generative quality.

5.3 Downstream Machine Learning Tasks

To further validate our image translation approach, we designed a downstream machine learning experiment to assess whether translated images can improve the performance of an object detection and classification model through data augmentation. Specifically, we evaluated the impact of using translated indoor soybean images

(converted into outdoor scenes) to complement real field images for weed detection and classification.

5.3.1 Datasets for Weed Detection and Classification

Three datasets were employed in this task:

- **Outdoor soybean field images.** A benchmark dataset consisting of 10,371 high-quality field images of soybean plants was used as the primary training and evaluation source [62]. These images capture natural variability in background, lighting, and occlusion conditions typical of real-world agricultural environments.

- **Indoor soybean plant images.** Approximately 68,000 images of soybean plants grown under controlled indoor conditions were available. These images, captured by the EAGL-I system [62], served as the basis for generating translated outdoor-like soybean images using our translation pipeline.

- **Weed images.** Cropped images of three common weed species—foxtail, pigweed, and kochia—were included. These images, also collected in the study by Beck et al. [62], were overlaid onto both real and translated soybean field images to construct labeled datasets for weed detection and classification.

5.3.2 Experiment Setup

Since the text-conditioning approach is less suitable for preserving the structural semantics of multiple objects within a single image[2], it was not ideal for the down-

stream machine learning task, where multiple soybean plants must be translated simultaneously from indoor to outdoor scenes. To address this, we adopted an image-based conditional generation strategy, as introduced in Chapter 3. This approach leverages the input image itself as structural guidance during the denoising process, ensuring that spatial arrangements and object-level consistency are retained while the environmental appearance is adapted to outdoor field conditions.

Since our indoor soybean dataset primarily consists of single plants imaged under controlled laboratory conditions, additional preprocessing steps were necessary to simulate realistic outdoor field scenes. Specifically, we constructed composite images that mimic the arrangement of multiple soybean plants in rows, as typically observed in field conditions. The overall experimental workflow consisted of the following stages:

1. **Creating outdoor soybean–weed images.** Cropped weed images were randomly sampled and overlaid onto the 10,371 outdoor soybean field images. Weed patch sizes were adjusted to realistic scales, and placement positions were chosen to avoid overlapping with soybean plants. Each soybean image was augmented with between one and three weeds.

2. **Cropping indoor soybean patches.** Indoor soybean images generally contain a single plant per image. To construct composite indoor field-like scenes, we first segmented and cropped soybean patches. Bounding boxes were obtained using YOLOv8 [73], and background pixels were removed using the color-thresholding approach implemented with PlantCV, as described in Chapter 4. Noisy patches (*e.g.*, fragmented plants, cropped leaves, or incomplete captures)

were removed through manual inspection, resulting in approximately 40,000 high-quality patches.

3. **Compositing indoor soybean patches onto a canvas.** Cropped soybean patches were arranged on empty canvases to resemble outdoor row structures. For each canvas, 3–6 patches were randomly selected, aligned either in rows or columns, and then randomly placed on the canvas. Outdoor background images without plants were inserted as the base layer, ensuring a more realistic compositional appearance.

4. **Translating composited indoor images into outdoor environments.** The composited soybean images were processed by an image-guided translation model (described in Chapter 3) to generate realistic outdoor scenes. Unlike the single-object case where DreamBooth was effective, it was less suitable here due to difficulties in capturing structural semantics of multiple scattered objects. Instead, we adopted an image-conditioned approach [23], using the composite images as direct guidance for the generation process. The details of indoor soybean images background-removal, cropping and composition are shown in Figure 5.4.

5. **Creating translated soybean–weed images.** Weeds were added to the translated soybean images following the same procedure used for real outdoor soybean images, ensuring consistency across datasets.

6. **Mixing outdoor soybean–weed images with translated images.** To evaluate the effect of translated data, we constructed mixed datasets by aug-

menting real outdoor soybean–weed images with translated counterparts. We experimented with synthetic ratios of 0%, 50%, 100%, 150%, 200%, and 300%. At 0%, only real images were used, while higher ratios progressively increased the proportion of translated images.

7. **Training an object detection and classification model.** For each weed species and synthetic ratio, we fine-tuned a pre-trained YOLOv8 model [73] for joint detection and classification of soybean and weed plants. YOLOv8 was chosen for its fast training speed and strong baseline performance for this task. Since our goal is to evaluate performance trends under different synthetic data ratios rather than achieve state-of-the-art accuracy, YOLOv8 provides a suitable benchmark, and its performance shows more noticeable improvement with added synthetic data compared to newer models like YOLOv11 [74].

8. **Evaluating model performance.** The fine-tuned YOLOv8 models were evaluated at different synthetic ratios using standard metrics:

    - **Precision:** the proportion of correctly identified positive samples (weeds in our case) among all predicted positives.
    - **Recall:** the proportion of correctly identified positive samples among all actual positives.
    - **mAP50:** mean average precision at an intersection-over-union (IoU) threshold of 0.5, reflecting the detection performance across all classes.

The real soybean–weed images were split into training and test sets with an 60/40 ratio. All YOLOv8 models were trained on the training split and evaluated on the

held-out test split which contains only real soybean-weed images.

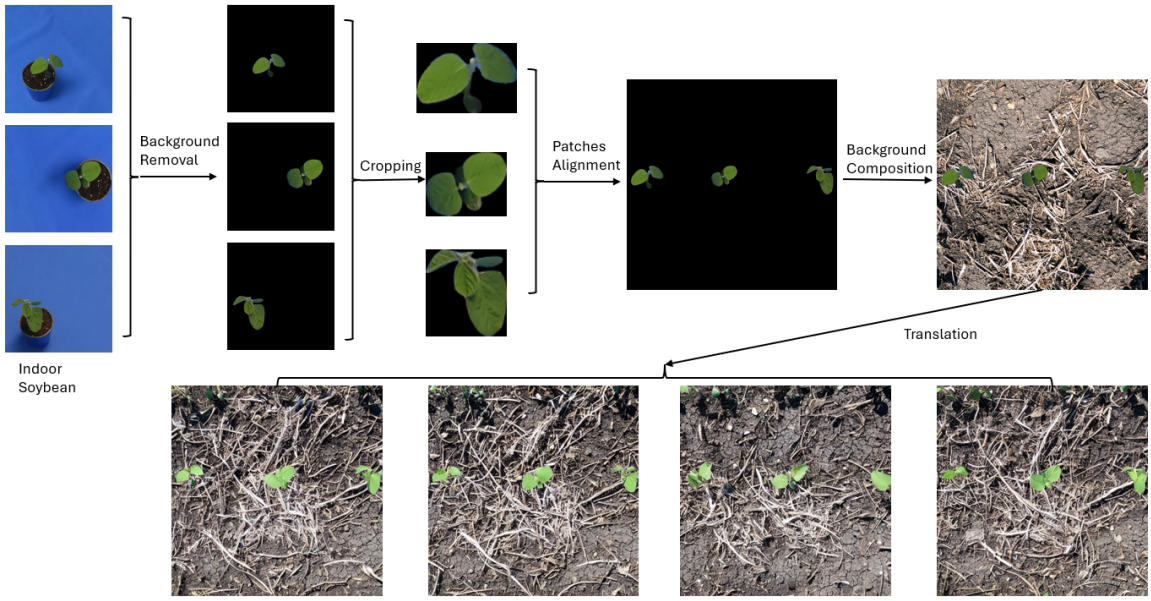

Figure 5.4: Overview of the dataset construction and translation workflow for the downstream indoor-to-outdoor soybean image translation. The process consists of: (1) removing background of the indoor soybean images, (2) cropping high quality plant patches, (3) compositing multiple patches onto a background canvas to mimic field-like arrangements, and (4) generating realistic outdoor translations using an image-conditioned diffusion model.

5.3.3 Image-Guided Translation Model

For the translation task, we employed the `img2img` pipeline of the Stable Diffusion framework [2; 23]. The input to this pipeline consisted of the composited indoor soybean images described in the previous subsection, which served as structural references for generating realistic outdoor field scenes. The Image-to-Image (`img2img`) pipeline is a conditional diffusion process designed to refine or translate an existing image under the guidance of a text prompt. Unlike unconditional text-to-image generation, which begins from random Gaussian noise, the `img2img` approach initializes

the denoising trajectory from a noised version of a given reference image. In our case, the reference was the composited indoor soybean images, and the conditioning prompts specified outdoor field characteristics. This setup allows the model to preserve the spatial structure and arrangement of the soybean plants while adapting environmental features such as lighting, soil texture, and background complexity to resemble outdoor conditions.

5.3.4 Evaluation and Results

Figure 5.5 provides a visual comparison between a real outdoor soybean image with weeds (left) and a translated soybean image generated from an indoor plant (right), where weeds were added in post-processing. The translated images closely resemble the real field images, with weeds successfully integrated into the scene.

To quantitatively assess performance, mixed datasets containing both real and translated soybean–weed images were used to train YOLOv8 detection and classification models at varying synthetic ratios. Each model was trained for 50 epochs, and the checkpoint with the lowest loss on the validation set was selected for evaluation.

Figure 5.6 shows an example of detection results for a model trained with a synthetic ratio of 300%. The left column presents the ground-truth labels, while the right column shows the corresponding predictions. The model successfully detects and classifies both soybean plants and weeds, achieving strong agreement with the ground truth. Interestingly, in some cases the model identified weeds that were not annotated in the training set—for instance, in the first-row example, where an unlabeled weed was correctly detected and classified. This suggests that the inclusion of

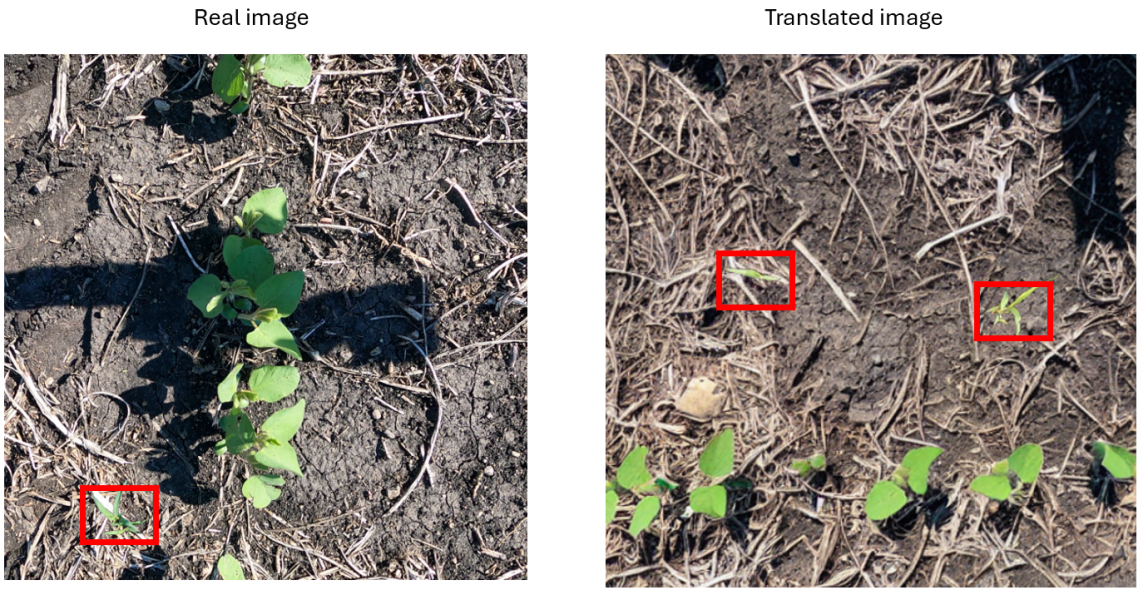

Figure 5.5: Comparison of real and translated soybean images with weeds. Left: a real outdoor soybean image with weeds superimposed on it. Right: a translated outdoor-like soybean image generated from an indoor plant, with weeds added. Weeds are labeled with red bounding boxes.

translated images may enhance the model's ability to generalize beyond the provided annotations.

Figure 5.7 and Table 5.1 summarize the performance of the fine-tuned YOLOv8 model across different synthetic ratios. Overall, we observe consistently high detection accuracy for all weed species, with mAP50 values averaging around 0.96. Both precision and recall tend to improve as the proportion of translated images in the training set increases.

For example, as shown in Figure 5.7, precision for the weed species *kochia* improves steadily from 0.911 with no synthetic augmentation to 0.938 at a synthetic ratio of 300%. This trend indicates that translated images serve as effective augmentations, enhancing the model's ability to detect and classify weeds under diverse conditions.

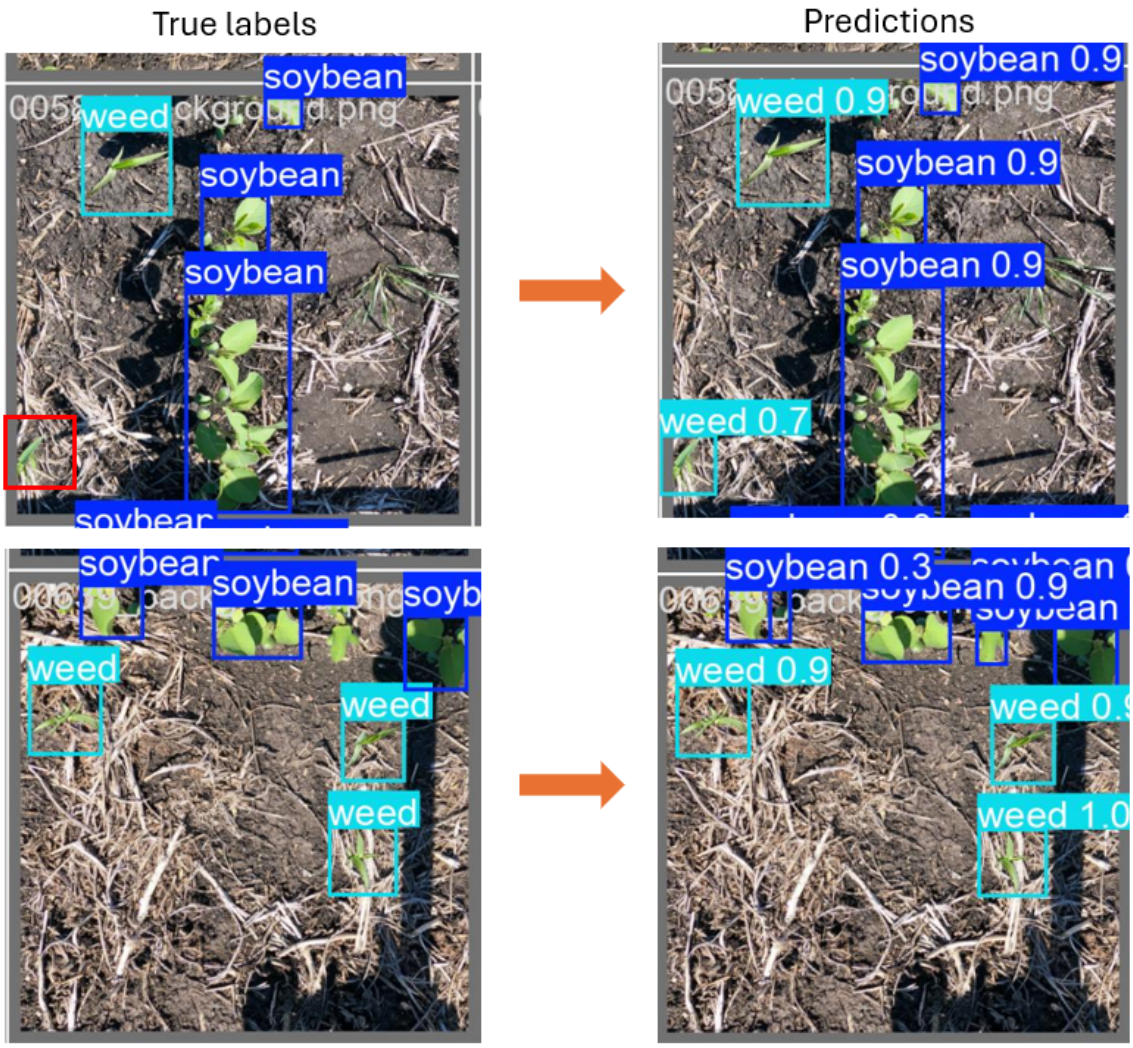

Figure 5.6: Detection and classification results of YOLOv8 trained with 300% synthetic ratio. Left: ground-truth annotations. Right: model predictions. The model successfully detects and classifies soybean and weed plants, and in some cases identifies weeds not included in the ground-truth labels.

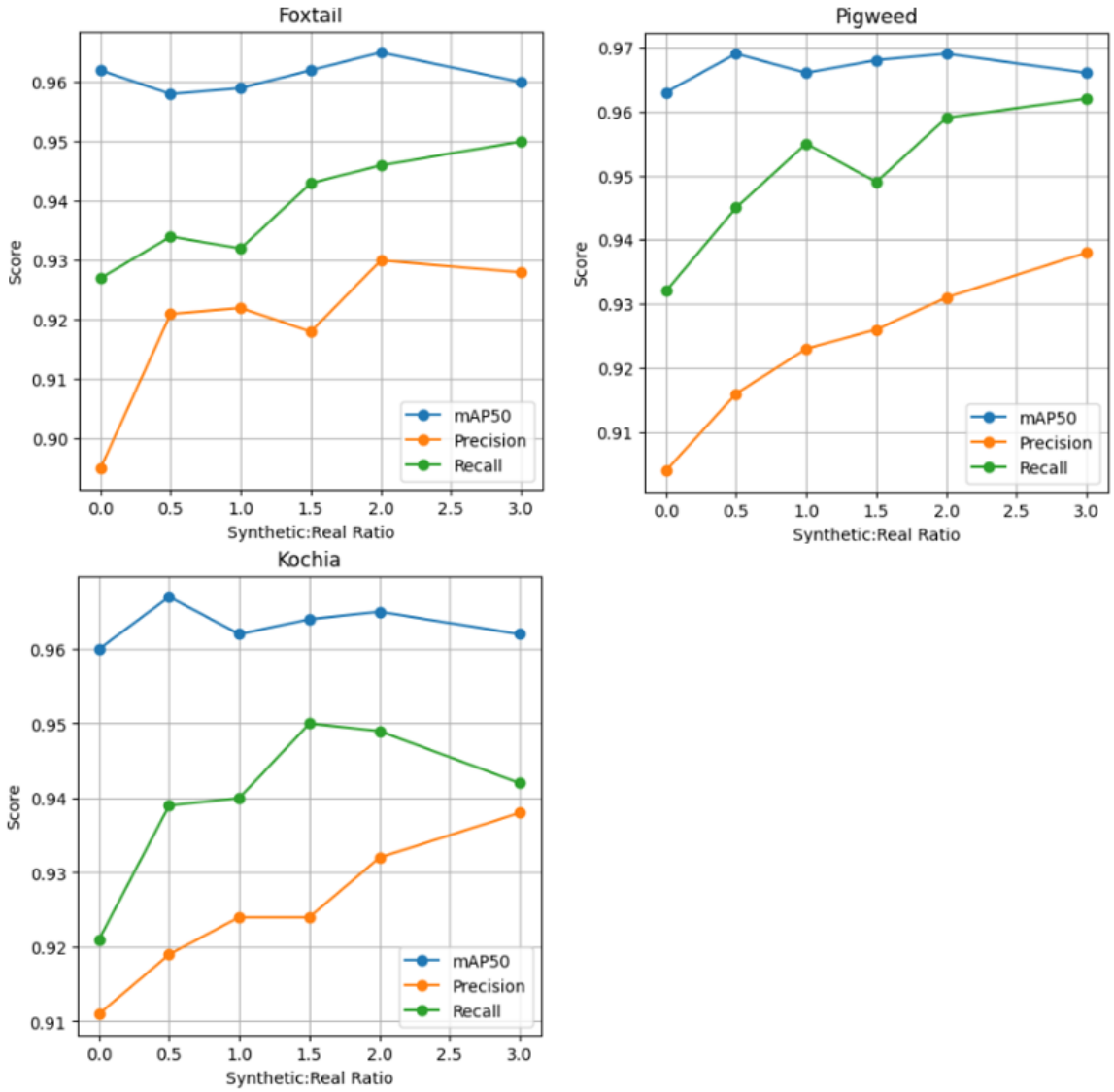

Figure 5.7: Performance of YOLOv8 on weed detection across synthetic ratios. Precision, recall, and mAP50 metrics are reported for multiple weed species. A steady improvement is observed with higher proportions of translated images.

5.4 Discussion

5.4.1 Strengths and Limitations of DreamBooth for Translating Agricultural Images

DreamBooth provided a powerful mechanism for text-conditioned translation by associating rare identifier tokens (*e.g.*, `sks`) with plant structures while condition-

Table 5.1: YOLOv8n performance on different synthetic ratios.

Ratio	Foxtail			Pigweed			Kochia		
	Precision	Recall	mAP50	Precision	Recall	mAP50	Precision	Recall	mAP50
0.0	0.895	0.927	0.962	0.904	0.932	0.963	0.911	0.921	0.960
0.5	0.921	0.934	0.958	0.916	0.945	0.969	0.919	0.939	0.967
1.0	0.922	0.932	0.959	0.923	0.955	0.966	0.924	0.940	0.962
1.5	0.918	0.943	0.962	0.926	0.949	0.968	0.924	0.950	0.964
2.0	0.930	0.946	0.965	0.931	0.959	0.969	0.932	0.949	0.965
3.0	0.928	0.950	0.960	0.938	0.962	0.966	0.938	0.942	0.962

ing on outdoor environment prompts. This approach offered strong controllability, enabling users to generate diverse outdoor scenarios by modifying textual inputs. For single-plant indoor images, DreamBooth successfully preserved plant morphology while embedding the subject into field-like contexts, making it particularly suitable for tasks where structural fidelity of individual plants is critical.

However, DreamBooth also revealed limitations in agricultural translation tasks. DreamBooth struggled with multi-object images: when applied to scenes containing multiple plants arranged in rows, it often failed to capture the spatial structure or scattered semantics accurately. This restricted its applicability to more complex field-level translations.

5.4.2 Strengths and Limitations of Image-Based Approaches for Translating Agricultural Images

In contrast, the image-guided `img2img` pipeline demonstrated strong performance in handling multi-object agricultural images. By initializing the denoising process from composited indoor soybean images, the model was able to preserve plant arrangements while adapting environmental features to outdoor conditions. This allowed the generation of realistic field scenes with multiple plants, a requirement for downstream tasks such as weed detection.

Nevertheless, the image-based approach is not without limitations. Its dependence on composite images means that preprocessing quality (segmentation, patch cropping, and background replacement) directly impacts the realism of the translated output. Artifacts such as imperfect boundaries or inconsistent lighting occasionally persisted into the final translated images. Moreover, compared to DreamBooth, `img2img` offers less fine-grained semantic control via prompts, which can limit flexibility for customized conditions.

5.4.3 Implications for Image Translation and Downstream ML Tasks

Several key lessons emerged from this study. First, the choice of translation strategy depends heavily on the target application. DreamBooth is well suited for controlled, single-object scenarios requiring high semantic precision, while image-guided translation is more effective for complex multi-object scenes representative of agri-

cultural fields. Future work may explore hybrid methods that combine text-inversion flexibility with image-guided robustness.

Second, the downstream experiments confirmed that translated images can meaningfully augment training data for machine learning tasks. Object detection models trained with both real and translated data achieved higher precision and recall, with clear improvements in recall for weed detection as the proportion of translated images increased. This demonstrates that translated data not only enriches visual diversity but also improves model generalization to real-world conditions.

Finally, these results underscore the importance of dataset design. High-quality segmentation, careful prompt engineering, and balanced mixing ratios of real and synthetic data were all crucial for success. Simply generating more synthetic data was not sufficient. Performance gains plateaued when the synthetic ratio became too large, suggesting that optimal augmentation requires a balance between real and translated images. In conclusion, these findings provide practical guidance for future work on agricultural image translation and its integration into downstream machine learning pipelines.

Chapter 6

Preference-Aligned Model Fine-tuning

6.1 Introduction

Recent advances in diffusion-based generative models have demonstrated remarkable capability in producing realistic and diverse images across a wide range of domains. Nevertheless, aligning these models with human preferences remains a challenging problem. While pretrained diffusion models can generate visually plausible images, the outputs are often misaligned with subtle semantic or aesthetic qualities valued by end-users. Scientific accuracy is key to creating synthetic data that can be used to solve real-world problems with machine learning. In applications such as digital agriculture, where fine-grained details of plants carry significant importance for downstream analysis, this misalignment can reduce the practical utility of generative models.

Preference alignment aims to bridge this gap by incorporating human feedback into the model refinement process. Conventional reinforcement learning approaches, such as reinforcement learning with human feedback (RLHF) [30], are powerful but computationally expensive and difficult to stabilize in high-dimensional image generation settings. As an alternative, supervised fine-tuning (SFT) approaches provide a more scalable framework for preference alignment [75; 76].

In this chapter, we present a modified version of the best-of-N supervised fine-tuning (Best-N-SFT) method [77; 78]. The core idea is to train a reward model on manually annotated data and subsequently use this reward model to score large batches of generated images. For each prompt, the model generates multiple candidate images, and the top-ranked subset is selected according to the reward model. These high-scoring candidates are then used to fine-tune the diffusion model through a weighted reconstruction loss, where the weights reflect the relative reward scores of the selected images. This process ensures that the model is explicitly encouraged to reproduce generations that align more closely with user preferences.

Our approach combines three critical components: (1) a manually curated dataset of expert-scored images to train a reliable reward model, (2) a reward predictor trained on latent representations from the VAE backbone of our fine-tuned Stable Diffusion model, and (3) a preference-weighted fine-tuning pipeline that leverages selective training. Together, these components yield a practical and efficient framework for aligning diffusion models with subjective human preferences, while maintaining training stability and scalability.

6.2 Materials and Methods

This study builds upon the image generative model described in Chapter 5, specifically a fine-tuned SD-1.4 text-to-image model trained to generate outdoor soybean images. The model was trained with the same parameter settings as outlined in Chapters 4 and 5. However, the model trained in Chapter 5 exhibited instability: under identical prompts, generated outputs varied significantly in quality due to the stochastic nature of the denoising process during inference. Examples of this variability are shown in Figure 6.1, where images generated from the same model and prompt differ in realism and fidelity.

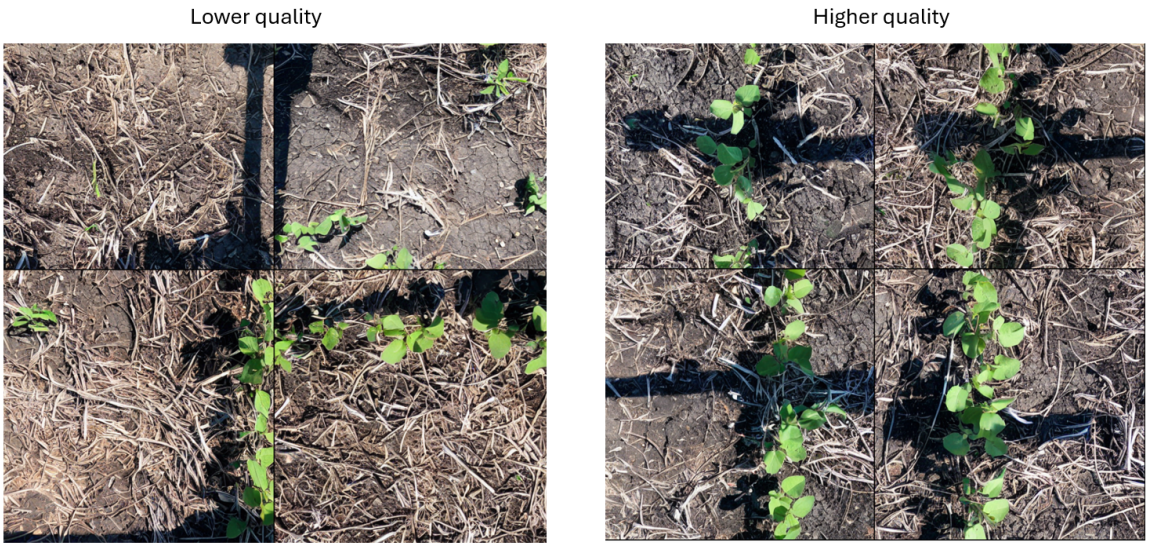

Figure 6.1: Examples of variability in image quality from the same model and prompt. Images on the left exhibit lower quality, while those on the right demonstrate higher-quality outputs, highlighting the quality variations of the output from baseline fine-tuned Stable Diffusion model.

6.2.1 Overall Experimental Setup

The objective of this study is to align a text-to-image diffusion model with human preferences through a two-stage pipeline that integrates manual feedback, reward modeling, and preference-weighted supervised fine-tuning:

1. **Preference annotation.** A pre-trained Stable Diffusion model fine-tuned on domain-specific plant imagery was used as the base generator. To capture human preferences, a pool of 500 images was generated under fixed hyperparameters of the image-to-image pipeline, including strength, guidance scale, and inference steps. Each image was manually scored by the user on a continuous scale from 0 to 1, based on subjective quality criteria such as realism, preservation of plant structure, and fidelity to the agricultural domain [1]. These annotations served as training data for the reward model.

2. **Reward modeling.** A CNN-based reward model was trained using the annotated images. Latent features were extracted from the encoder of the Stable Diffusion VAE, which compresses each image into a $4 \times 64 \times 64$ latent representation. Operating in the same latent space as the U-Net denoiser ensures that preference signals can be transferred consistently to guide the diffusion process.

3. **Preference-aligned fine-tuning.** The diffusion model was further fine-tuned using a Best-of-N supervised strategy modified with reward weighting. A total of 5,000 images were generated from diverse prompts, with $N = 12$ candidates produced per prompt. The reward model assigned scores to each candidate,

[1] Even though we used the term "expert-feedback" above, I scored these images as a proof of concept. In the future, we will collaborate with experts to score the images.

and the top $k = 8$ images were selected as pseudo-targets. A reward-weighted reconstruction loss was then computed, assigning higher weights to samples with stronger preference scores. This biased the parameter updates of the U-Net toward user-preferred outcomes while maintaining the efficiency of supervised fine-tuning.

An overview of the workflow is shown in Figure 6.2. The experimental design integrates manual scoring, reward modeling, and preference-weighted fine-tuning to progressively align the generative model with user preferences. For convenience, we refer to the Stable Diffusion model trained in Chapter 5 as the base model, and to the preference-aligned version as the user-preferred model, as illustrated in Figure 6.2.

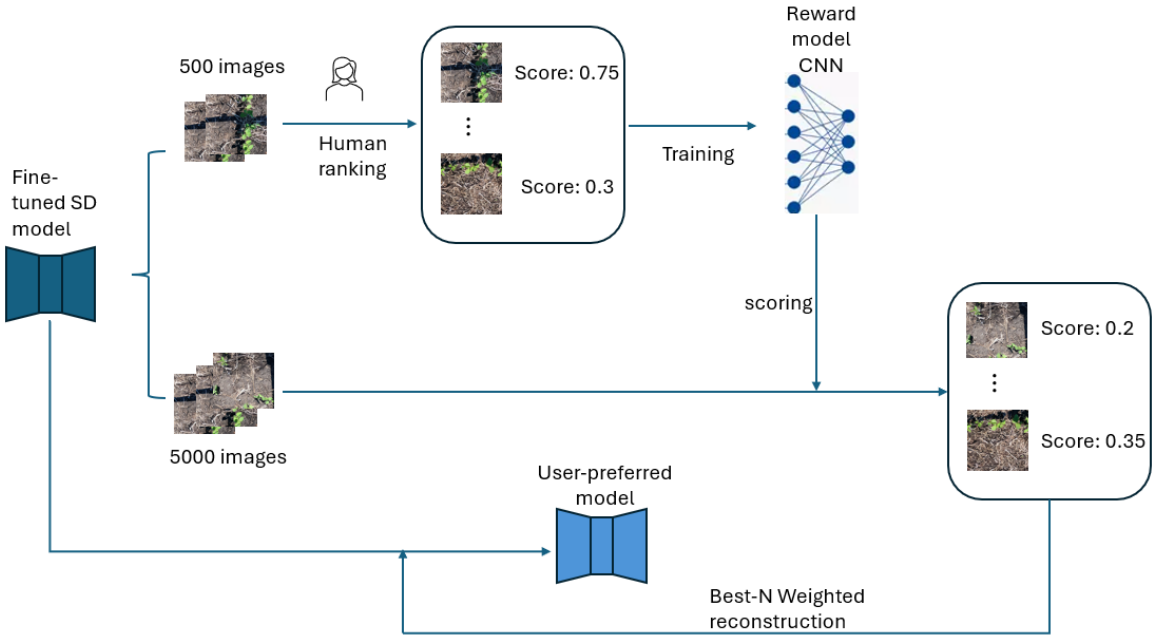

Figure 6.2: Overview of the preference-aligned fine-tuning pipeline. The workflow consists of three stages: (1) manual scoring of generated images, (2) training a CNN-based reward model in the latent space of the SD VAE, and (3) reward-weighted supervised fine-tuning using a Best-of-N strategy.

6.2.2 Reward Model

To capture user preferences and provide a feedback signal for fine-tuning, we trained a reward model on manually scored images. The model predicts a scalar reward $\hat{r} \in \mathbb{R}$ from the latent representation of an image, where the ground-truth reward $r \in [0, 1]$ corresponds to the user-assigned quality score.

Input Representation

Given an image $x \in \mathbb{R}^{3\times512\times512}$, we first encode it using the Stable Diffusion VAE. The encoder $E(\cdot)$ maps the image into a latent representation:

$$z = \alpha \cdot E(x) \in \mathbb{R}^{4\times64\times64}, \tag{6.1}$$

where $\alpha = 0.18215$ is the scaling factor used in Stable Diffusion to ensure that the variance of the latent space matches that of the U-Net denoiser. The reward model operates directly on this latent representation, thereby maintaining consistency with the diffusion model's generative process.

Model Architecture

The reward model is implemented as a CNN that progressively reduces the spatial dimensions of the latent tensor while increasing feature channels. Feature aggregation is achieved through convolutional blocks with batch normalization and ReLU activations, followed by global average pooling. The final fully connected head maps the aggregated features to a scalar value representing the predicted reward. The full architecture is summarized in Table 6.1.

Table 6.1: Architecture of the CNN-based reward model. The input is a latent tensor of shape $4 \times 64 \times 64$.

Layer	**Output Shape**	**Details**
Input	$4 \times 64 \times 64$	Scaled VAE latent
Conv2d + BN + ReLU	$32 \times 32 \times 32$	$4 \rightarrow 32$, kernel=3, stride=1, pool=2
Conv2d + BN + ReLU	$64 \times 16 \times 16$	$32 \rightarrow 64$, kernel=3, stride=1, pool=2
Conv2d + BN + ReLU	$128 \times 8 \times 8$	$64 \rightarrow 128$, kernel=3, stride=1, pool=2
Conv2d + BN + ReLU	$256 \times 1 \times 1$	$128 \rightarrow 256$, kernel=3, stride=1, global avg pool
Flatten	256	–
Linear + ReLU + Dropout	128	Dropout $p = 0.3$
Linear	1	Reward prediction

Training Procedure

A dataset of $N = 500$ generated images was collected, with each image manually annotated with a preference score $r \in [0, 1]$. The dataset was randomly split into training (70%) and validation (30%) subsets. The reward model was optimized using the AdamW optimizer with a learning rate of 1×10^{-4} and trained to minimize the mean squared error loss (MSE):

$$\mathcal{L}_{\text{reward}} = \frac{1}{N} \sum_{i=1}^{N} (\hat{r}_i - r_i)^2 , \tag{6.2}$$

where $\hat{r}_i$ is the predicted reward and r_i is the ground-truth user-assigned score. During training, the Pearson correlation coefficient between predicted and ground-truth rewards was monitored to evaluate alignment with user preferences.

Results of the Reward Model

Figure 6.3 illustrates the training dynamics of the reward model (panel a) and its performance on the test set (panel b). The predicted reward scores showed strong agreement with human-annotated scores, achieving a Pearson correlation of 0.791. This result indicates that the reward model effectively captures visual features associated with higher user preference, thereby encoding a bias aligned with subjective human evaluation.

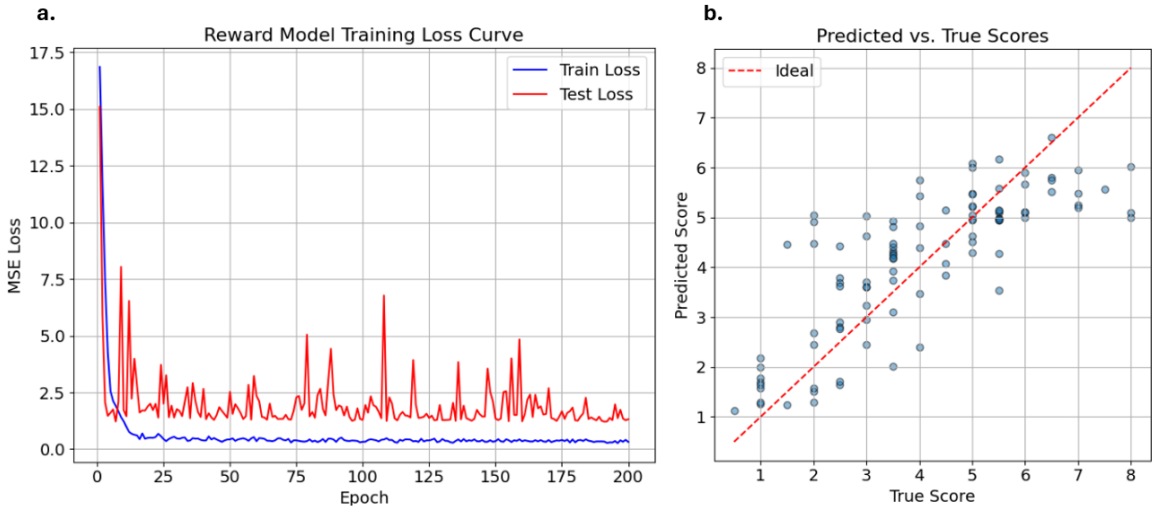

Figure 6.3: Performance of the reward model. (a) Training dynamics of the CNN-based reward model. (b) Predicted reward scores versus ground-truth user scores on the test set, showing a Pearson correlation of 0.791. These results confirm that the reward model successfully encodes user preferences into a predictive signal.

Usage in Fine-Tuning

Once trained, the reward model was frozen and integrated into the preference-aligned fine-tuning process of the diffusion model. For each prompt, the diffusion model generated multiple candidate images, which were then scored by the reward model. The scores were used to construct a reward-weighted reconstruction loss, ensuring that higher-scoring images contributed more strongly to the U-Net updates. In this way, the reward model served as a fixed guide, transferring user preferences into the optimization process and aligning the diffusion model with subjective quality criteria.

6.2.3 Fine-Tuning with Reward Model Guidance

After training the reward model, it was integrated into the fine-tuning stage of the Stable Diffusion model we trained in Chapter 5.

Candidate Generation and Scoring

During fine-tuning, a candidate pool of synthetic images was generated using the base model. For each prompt p, $N = 12$ candidate images were sampled:

$$\{x_1, x_2, \ldots, x_N\} \sim \mathcal{G}_\theta(p), \tag{6.3}$$

where $\mathcal{G}_\theta$ denotes the generative model parameterized by θ. Each candidate image was then encoded into its latent representation z_i using the VAE encoder, and scored by the reward model:

$$r_i = f_\phi(z_i), \quad i = 1, \ldots, N, \tag{6.4}$$

where f_ϕ is the frozen reward model. The top-k candidates (with $k = 8$) were selected according to their reward scores and used for optimization.

Reward-Weighted Reconstruction Loss

For the selected top-k images, the standard denoising reconstruction loss of the diffusion model was modified with reward-based weighting. To emphasize higher-scoring samples, the weights were computed using a softmax-normalized distribution:

$$w_i = \frac{\exp(r_i/\tau)}{\sum_{j=1}^{k} \exp(r_j/\tau)}, \quad i = 1, \ldots, k, \tag{6.5}$$

where τ is a temperature hyperparameter controlling the sharpness of the weighting distribution. The final training objective was defined as:

$$\mathcal{L}_{\text{SFT}} = \sum_{i=1}^{k} w_i \left\| \epsilon_\theta(z_t^i, t) - \epsilon \right\|^2, \tag{6.6}$$

where ϵ_θ is the U-Net's predicted noise, ϵ is the true Gaussian noise, and z_t^i is the noisy latent sampled at timestep t using the diffusion forward process. This formulation biases learning toward higher-scoring candidates, ensuring that parameter updates are preferentially guided by user-aligned outcomes.

Training Setup

In total, 5,000 candidate images were generated across diverse prompts. For each training step, $N = 12$ candidates were produced, and the top $k = 8$ were selected for

optimization. The U-Net denoiser was fine-tuned using the AdamW optimizer with a learning rate of 1×10^{-5}, and gradient clipping set to 0.5. An exponential moving average of model weights was maintained with a decay factor of 0.999 to improve training stability. Validation was performed at the end of each epoch by generating $M = 50$ images from a held-out set of prompts. These images were scored by the reward model, and the mean reward score was recorded. The best model checkpoint was selected based on the highest validation mean reward.

6.2.4 Evaluation and Results

The performance of the model was evaluated throughout the fine-tuning process. At the end of each epoch, the model was prompted to generate a set of 50 images, which were subsequently scored by the reward model. The average reward score across these images was recorded as an evaluation metric. As shown in Figure 6.4, the mean reward score steadily increased with training epochs, indicating progressive alignment with user preferences. The best-performing model, corresponding to epoch 17, was selected as the final fine-tuned model.

In addition, we compared the user-preferred model against the base model in terms of mean reward score and FID. As summarized in Table 6.2, the tuned model achieved a substantially higher mean reward score (6.884 vs. 3.722 for the base model). A paired t-test confirmed that this difference was statistically significant ($t = 10.98$, $p = 1.42 \times 10^{-17}$). FID values were also computed relative to real outdoor soybean images. Interestingly, the user-preferred model obtained a higher FID score (189.33 vs. 108.76), suggesting that while the outputs were more aligned with human

preferences, they deviated further from the distribution of real images according to FID. This highlights the difference between objective generative quality metrics and subjective preference alignment.

Qualitative results are shown in Figure 6.5. Images generated by the base model often contained fewer soybean plants, with some leaves appearing distorted or unrealistic. In contrast, the tuned user-perferred model produced more stable outputs, with plants centered in the frame and leaf structures that more closely resembled real outdoor soybean images.

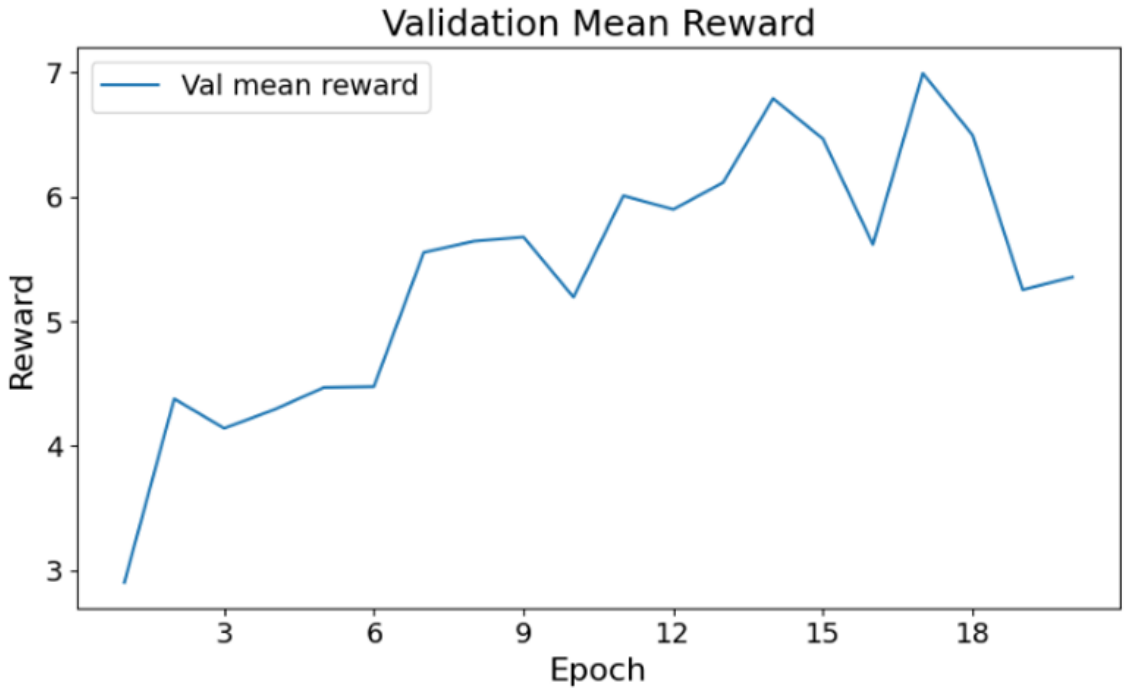

Figure 6.4: Reward evaluation during fine-tuning. The mean reward score, averaged across 50 generated images per epoch, increases steadily with training. The best-performing model, obtained at epoch 17, was selected as the final tuned model.

Table 6.2: Comparison of our trained base Stable Diffusion model and the preference-aligned model. Higher reward scores indicate better alignment with user preferences, while lower FID values indicate closer similarity to the real image distribution.

Model	**Mean Reward Score**	**FID (vs. Real)**
Base Model	3.722	108.76
Preference-aligned Model	6.884	189.33

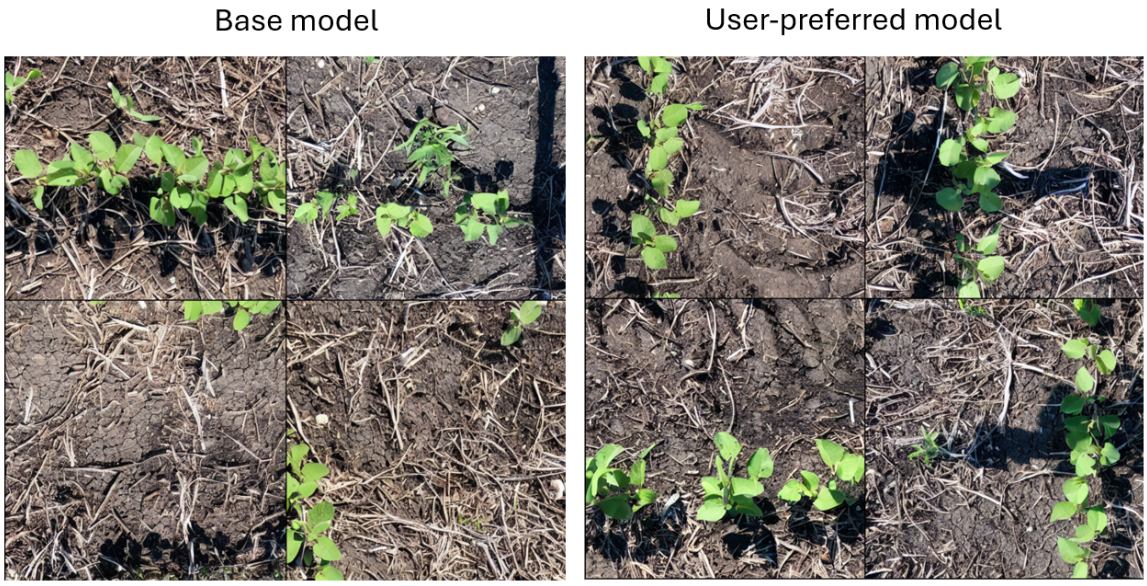

Figure 6.5: Qualitative comparison of generated images. Left: examples from the base model, showing fewer plants and structural distortions in leaves. Right: examples from the preference-aligned user-preferred model, producing more stable, centered plants with leaf details more consistent with real outdoor soybean imagery.

6.3 Discussion

The experiments in this chapter investigated a preference-aligned fine-tuning strategy for diffusion models, guided by a learned reward model trained on user-annotated data. By leveraging a modified Best-N-SFT approach, the model successfully adapted to align image generation more closely with user preferences. The results highlight several important insights and implications.

6.3.1 Performance Improvements

The preference-aligned model achieved a significantly higher mean reward score compared to the base model (6.884 vs. 3.722), indicating that the fine-tuning procedure effectively improved the alignment between generated images and user-defined quality criteria. This demonstrates the feasibility of integrating a reward model into the training loop to steer generation toward subjective notions of quality.

6.3.2 Tradeoffs and Limitations

While the reward scores improved, the FID deteriorated when comparing the tuned model to real images (189.33 vs. 108.76 for the base model). This tradeoff suggests that optimizing for user preference alignment can shift the generative distribution away from the statistical characteristics of real data. Such divergence is a common phenomenon in preference learning, reflecting a domain shift from the base model to the preference-aligned model. Importantly, this shift may not be entirely detrimental: fine-tuning can reduce the prevalence of lower-quality outputs, effectively concentrating the distribution toward samples that better align with user-defined notions of quality. Moreover, the reliance on a reward model introduces potential bias, as the reward predictor is limited by the scope, quality, and consistency of the user-annotated training set. The Pearson correlation between predicted and true scores, though moderately high, suggests room for improvement in the reward model's reliability.

6.3.3 Practical Implications

Despite these limitations, the proposed approach provides a practical framework for incorporating subjective expert feedback into diffusion model training without requiring paired supervision. This paradigm is particularly valuable in scientific application domains such as agriculture and medicine, where "ground truth" image quality is inherently subjective or difficult to quantify. The ability to guide generative models using preference data opens opportunities for tailoring outputs to specific user groups or domain experts.

6.3.4 Future Directions

Future research can extend this work by improving reward model architectures, experimenting with reinforcement learning or differentiable ranking objectives, and exploring strategies to balance preference alignment with realism. Additionally, larger and more diverse user-annotated datasets will be crucial for improving the robustness and generalizability of reward-based fine-tuning. Finally, hybrid methods that combine supervised signals with reward-driven optimization may help reduce the observed tradeoffs between fidelity and alignment.

6.3.5 Summary

Overall, the reward-model-guided fine-tuning improved subjective alignment with user preferences, but it may be at the cost of decreased distributional fidelity to real images. This highlights both the promise and the challenges of preference-based generative model tuning, and motivates further exploration of methods that balance

alignment with realism.

Chapter 7

General Discussion and Conclusion

7.1 Overview of Contributions

This thesis set out to address a central challenge in agricultural artificial intelligence: the scarcity of large, and annotated datasets for training robust machine learning models. Plant phenotyping, crop monitoring, and weed detection tasks all rely on extensive image collections, yet the acquisition of field data is expensive, time-consuming, and heavily constrained by environmental variability. Generative models offer a promising avenue for overcoming these limitations by synthesizing realistic plant images that can complement existing datasets. The contributions of this work span three complementary directions, each corresponding to a major chapter of this thesis:

1. **Text-to-image generation (Chapter 4).** A SD-v1.4 model was fine-tuned with domain-specific datasets of indoor and outdoor canola imagery. By conditioning on captions generated using large language models, the system was able

to produce realistic images of plants under a wide range of growth stages and conditions. The generated images were evaluated quantitatively using FID and IS metrics, as well as qualitatively through visual inspection. In addition, the utility of synthetic data for downstream classification tasks was validated using benchmark datasets, demonstrating that generative augmentation can improve phenotype classification accuracy.

2. **Indoor-to-outdoor image translation (Chapter 5).** While indoor images are more abundant and easier to capture under controlled conditions, they differ significantly from field imagery in background, lighting, and variability. To bridge this gap, two complementary translation strategies were explored: DreamBooth-based text conditioning and image-guided diffusion using the `img2img` pipeline. The former offered semantic control through rare-token association, while the latter preserved multi-object spatial structures necessary for realistic field scenes. The translated images were subsequently applied to a weed detection and classification task, where mixing real and translated data improved YOLOv8 performance across multiple synthetic ratios.

3. **Preference-aligned fine-tuning (Chapter 6).** Given that fine-tuned Stable Diffusion models often produce diverse yet unstable image quality, this chapter introduced a preference-guided fine-tuning framework. A reward model was trained on manually annotated images to predict user preference scores within the latent space of Stable Diffusion. This reward model was then incorporated into a reward-weighted supervised fine-tuning procedure, ensuring that higher-scoring images exerted greater influence on model updates. The

resulting preference-aligned model achieved higher reward scores than the base model, yielding more stable and expert-aligned outputs despite trade-offs in conventional objective metrics such as the FID score.

Taken together, these contributions demonstrate a systematic exploration of generative modeling in agriculture: from adapting text-to-image models for domain-specific generation, to translating indoor imagery into outdoor scenes, to aligning models with subjective human preferences. This integrated pipeline not only expands the availability of training data but also enhances the usability and flexibility of synthetic imagery in downstream agricultural applications.

7.2 Synthesis of Findings

This thesis explored diffusion-based generative pipelines across three complementary tasks: text-to-image generation, indoor-to-outdoor translation, and preference-aligned fine-tuning. Although each chapter addressed different challenges, several broader findings emerge when the results are considered together.

7.2.1 Effectiveness of Diffusion Models for Agricultural Imagery

Across all experiments, diffusion models demonstrated strong potential for generating high-quality plant imagery in agricultural domains. Fine-tuning Stable Diffusion on curated datasets of indoor and outdoor plant images enabled the production of synthetic crops that were visually realistic and semantically meaningful. These im-

ages were not only valuable for visualization, but also proved effective for augmenting real datasets in downstream machine learning tasks. For instance, in phenotype classification, synthetic images improved model accuracy when mixed with real data, particularly at higher augmentation ratios.

The translation experiments further highlighted the versatility of diffusion models. By adapting indoor soybean imagery to outdoor conditions, translated images enriched training sets for weed detection and classification. This allowed models such as YOLOv8 to achieve higher precision and recall, especially at moderate synthetic ratios, underscoring the ability of generative models to bridge gaps in data collection.

Finally, preference-guided fine-tuning improved subjective quality and stability of generated outputs. By leveraging a reward model trained on expert annotations, the diffusion process was explicitly biased toward expert-valued attributes such as plant structure fidelity and visual realism. This approach produced outputs that were more consistent across inference runs, addressing the instability observed in the baseline fine-tuned models.

7.2.2 Trade-offs Between Objective Metrics and Subjective Quality

A recurring theme across the chapters was the divergence between objective image quality metrics and subjective human evaluation. While FID and IS provided quantitative benchmarks for generative quality, they did not always align with perceived realism or usability. In several cases, models that performed well according to FID or IS produced images that appeared less convincing to experts.

Preference-aligned fine-tuning, on the other hand, improved outputs as judged by expert scorers, but at the cost of worse FID scores. This discrepancy highlights an important consideration for AI in agriculture: objective metrics alone are insufficient for evaluating generative pipelines. Instead, a multi-dimensional evaluation strategy is needed—one that combines quantitative similarity measures with subjective preference alignment and downstream task performance. This ensures that synthetic data is not only statistically close to real data, but also practically useful and aligned with expert judgments.

7.2.3 Lessons Learned Across Chapters

Several lessons can be summarized from the progression of Chapters 4 to Chapter 6.

First, in text-to-image generation (Chapter 4), effective augmentation depends heavily on the quality of captions and preprocessing. Accurate, semantically rich text descriptions were essential for conditioning diffusion models to produce useful plant imagery.

Second, in indoor-to-outdoor translation (Chapter 5), different methods offered distinct trade-offs. DreamBooth excelled at controlling semantic associations for single plants but struggled with multi-object scenes, while the image-guided `img2img` approach better preserved structural layouts in field-like settings. This suggests that translation strategies must be chosen based on the complexity of the target imagery.

Third, preference-aligned fine-tuning (Chapter 6) demonstrated that human feedback can be incorporated into generative training through reward modeling. This of-

fers more adaptive and user-centered alignment of outputs from the generative models. Such methods may be particularly important in agricultural contexts, where expert knowledge and subjective judgment often guide decisions about image realism and utility.

Taken together, these findings illustrate both the promise and complexity of applying diffusion models in agriculture. They reveal that success depends not only on model architecture and training data, but also on the careful design of work pipelines, image translation strategies, and output alignment with expert preferences.

7.3 Practical Implications for AI in Agriculture

The results of this thesis have several practical implications for the development and deployment of artificial intelligence systems in agriculture.

First, the ability to generate and translate plant images directly addresses one of the largest barriers in agricultural machine learning: the scarcity of annotated field datasets. Collecting large-scale outdoor imagery is labor-intensive, seasonally constrained, and often requires costly expert annotation. By leveraging generative diffusion models, synthetic images can supplement real data, effectively reducing reliance on exhaustive field collection efforts. This not only lowers the cost of dataset development but also accelerates the pace of AI experimentation and deployment in agricultural research.

Second, the experiments confirmed that synthetic augmentation improves the performance of downstream machine learning tasks. In phenotype classification, mixing generated images with real data improved classification accuracy across multiple crop

species. Similarly, in weed detection and classification, translated indoor-to-outdoor images enhanced YOLOv8's precision and recall, particularly at moderate augmentation ratios. These findings demonstrate that generative data can contribute directly to the robustness and generalization of computer vision models in agricultural settings.

Finally, preference-aligned fine-tuning introduces a new dimension of adaptability to generative systems. By integrating expert feedback through a reward model, outputs can be steered toward expert-defined criteria such as plant morphology, structural fidelity, or botanic realism. This form of alignment enables generative pipelines to move beyond generic image synthesis and become practical tools that reflect the nuanced requirements of agricultural experts. In practice, such adaptability could facilitate automated quality control in plant monitoring, precision agriculture, and digital phenotyping workflows.

In summary, these contributions show that diffusion-based generative models, when combined with translation and preference-alignment strategies, have the potential to transform how agricultural datasets are built, extended, and applied in downstream AI systems.

7.4 Limitations and Challenges

Despite the promising results presented in this thesis, several limitations and challenges remain that must be acknowledged.

First, the quality of generated and translated images exhibited variability across experiments. While many outputs were highly realistic, others contained visual artifacts such as distorted leaves, unnatural lighting, or inconsistent backgrounds. This

variability reflects the inherent stochasticity of the diffusion process as well as the difficulty of perfectly modeling complex field environments. For practical deployment, additional refinement steps such as post-processing or reinforcement learning with expert feedback, may be required to ensure consistent image quality.

Second, trade-offs were observed between subjective preference alignment and objective similarity metrics. Preference-aligned fine-tuning improved outputs as judged by experts, yet it resulted in higher FID scores, suggesting a greater divergence from the statistical distribution of real images. This highlights a broader methodological challenge: objective metrics alone are insufficient for evaluating generative systems, while purely subjective evaluation lacks scalability. Balancing these two perspectives remains an open research problem.

Third, the scalability of manual preference annotation is limited. The reward model in Chapter 6 relied on expert scoring of generated images, which is time-consuming and may introduce annotator bias. While effective at a small scale, this approach may not be feasible for training larger or more generalizable reward models. Future work will need to explore active learning or other RLHF strategies to reduce reliance on extensive manual labeling.

Additionally, we observed that some generated images show plant shadows that look inconsistent as shown in Figure 5.2. This suggests the model is not fully capturing the lighting conditions during training, resulting in shadows that look less physically plausible. In future work, we plan to evaluate plant shadows more systematically and explore ways to include information about the light source from the metadata, such as its direction or strength, when conditioning the model. This may

help produce shadows that are more physically coherent.

Finally, the methods presented in this thesis were primarily developed and tested on canola and soybean imagery. Although these crops are representative and agriculturally important, domain-specific biases may limit the generalization of the approaches to other species with different growth patterns, phenotypes, or environmental contexts. Broader validation across multiple crop types and regions will be necessary to establish the robustness of the proposed methods.

In summary, while this thesis demonstrates the feasibility and value of generative diffusion models in agriculture, overcoming these challenges will be essential for translating the methods into scalable and reliable tools for the agricultural AI community.

7.5 Future Directions

The findings of this thesis open several promising directions for future research in the application of generative diffusion models to agricultural imagery.

First, larger multi-modal models such as vision–language transformers could be explored for captioning and translation tasks. In this thesis, captioning relied on models such as ChatGPT-4o and Llava for generating descriptive prompts. Future work could incorporate state-of-the-art multi-modal architectures that jointly learn from image–text pairs at scale, thereby improving the semantic richness and accuracy of captions used for conditioning diffusion models.

Second, hybrid approaches that combine the strengths of different translation methods requires exploration. DreamBooth provides precise semantic control for

single-object images, while image-guided diffusion excels at handling multi-object, field-like scenes. A framework that unifies these approaches could deliver both fine-grained semantic fidelity and robustness in complex agricultural environments, enabling more versatile translation pipelines.

Fourth, the framework should be extended to a broader range of crops, phenotypes, and agronomic tasks. Future work could include applications in stress detection (*e.g.*, drought, nutrient deficiency, or pest damage), disease monitoring, and even yield prediction from synthetic field imagery. Such extensions would help evaluate the generalizability of the methods and strengthen their value across different agricultural domains.

Finally, integrating generative models with human-informed notions of botanical realism offers a promising avenue for model refinement and controllability. For instance, aligning diffusion models with structured crop knowledge could provide more precise guidance over generated outputs. Such integration would not only enhance realism but also ensure consistency with biological and environmental constraints, thereby making generative tools more reliable for agricultural applications.

7.6 Conclusion

This thesis has presented a comprehensive exploration of diffusion-based generative modeling for agricultural imagery, addressing key challenges in data scarcity, domain adaptation, and alignment with human preferences. The contributions span three major components. First, a Stable Diffusion model was fine-tuned to generate plant images from text prompts, demonstrating that synthetic imagery can aug-

ment real datasets and improve phenotype classification accuracy. Second, indoor-to-outdoor image translation methods were developed using both DreamBooth and image-guided pipelines, enabling the adaptation of abundant indoor imagery into field-like conditions that enriched weed detection datasets. Third, a preference-aligned fine-tuning framework was introduced, where a reward model trained on expert annotations guided the diffusion process toward outputs that better reflect expert judgments of quality and realism.

Together, these contributions address the fundamental limitation of agricultural AI: the limited availability of high-quality, annotated field data. By leveraging generative pipelines, this work shows that it is possible not only to expand the scale and diversity of training data, but also to improve the robustness of downstream machine learning models for plant phenotypes classification and weeds detection. Moreover, by incorporating expert-centered alignment, the thesis demonstrates a pathway for building generative systems that reflect expert-defined standards of utility.

The broader vision emerging from this work is that generative models can serve as a foundation for data-efficient, expert-aligned agricultural AI systems. As models continue to improve in scale and adaptability, their integration with domain knowledge and expert feedback will enable the creation of synthetic datasets that are not only statistically robust but also biologically and botanically meaningful. In doing so, generative pipelines have the potential to accelerate innovation in crop phenotyping, precision agriculture, and food security research, contributing to more sustainable and precise agricultural practices in the future.

Bibliography

[1] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[2] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 10 684–10 695.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[4] M. Minervini, A. Fischbach, H. Scharr, and S. A. Tsaftaris, "Finely-grained annotated datasets for image-based plant phenotyping," *Pattern recognition letters*, vol. 81, pp. 80–89, 2016.

[5] S. Bhugra, S. Srivastava, V. Kaushik, P. Mukherjee, and B. Lall, "Plant data generation with generative ai: an application to plant phenotyping," *Applications of Generative AI*, pp. 503–535, 2024.

[6] A. E. Krosney, P. Sotoodeh, C. J. Henry, M. A. Beck, and C. P. Bidinosti, "Inside out: transforming images of lab-grown plants for machine learning applications in agriculture," *Frontiers in Artificial Intelligence*, vol. 6, p. 1200977, 2023.

[7] C. Wang, Y. Xia, L. Xia, Q. Wang, and L. Gu, "Dual discriminator gan-based synthetic crop disease image generation for precise crop disease identification," *Plant Methods*, vol. 21, no. 1, p. 46, 2025.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[9] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *International Conference on Learning Representations (ICLR)*, 2018.

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[11] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning*. pmlr, 2015, pp. 2256–2265.

[12] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[13] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[14] D. Saxena and J. Cao, "Generative adversarial networks (gans) challenges, solutions, and future directions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–42, 2021.

[15] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 3836–3847.

[16] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *International conference on machine learning*. Pmlr, 2021, pp. 8821–8831.

[17] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML)*. PMLR, 2021, pp. 8748–8763.

[19] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dream-

booth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 22 500–22 510.

[20] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[21] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, "An image is worth one word: Personalizing text-to-image generation using textual inversion," *arXiv preprint arXiv:2208.01618*, 2022.

[22] N. Tumanyan, M. Geyer, S. Bagon, and T. Dekel, "Plug-and-play diffusion features for text-driven image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1921–1930.

[23] C. Meng, Y. Song, J. Song, J. Zhao, S. Ermon, and B. Poole, "Sdedit: Guided image synthesis and editing with stochastic differential equations," *arXiv preprint arXiv:2108.01073*, 2021.

[24] J. Shi, W. Xiong, Z. Lin, and H. J. Jung, "Instantbooth: Personalized text-to-image generation without test-time finetuning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 8543–8552.

[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with

conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[26] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2849–2857.

[27] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[28] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-image diffusion models," in *ACM SIGGRAPH 2022 conference proceedings*, 2022, pp. 1–10.

[29] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5505–5514.

[30] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *arXiv preprint arXiv:2203.02155*, 2022.

[32] S. Black, J. Menick, P. Krähenbühl, A. Botev, A. Jolicoeur-Martineau, R. Hennequin, B. Ganev, M. Caccia *et al.*, "Training diffusion models with reinforcement learning," *arXiv preprint arXiv:2306.00367*, 2023.

[33] K. Lee, R. Salakhutdinov, H. Zhang, Y. Sharma *et al.*, "Aligning text-to-image models using human feedback," *arXiv preprint arXiv:2302.12192*, 2023.

[34] K. Clark, P. Vicol, K. Swersky, and D. J. Fleet, "Directly fine-tuning diffusion models on differentiable rewards," *arXiv preprint arXiv:2309.17400*, 2023.

[35] R. Rafailov, Y. Sharma, E. Mitchell, A. Liu, A. Donsker *et al.*, "Direct preference optimization: Your language model is secretly a reward model," *arXiv preprint arXiv:2305.18290*, 2023.

[36] P.-H. Yeh, K.-H. Lee, and J.-C. Chen, "Training-free diffusion model alignment with sampling demons," *arXiv preprint arXiv:2410.05760*, 2024.

[37] G. G. Anil, D. M. Nagaraj, K. Shanmugam, and S. Shakkottai, "Rejection sampling based fine tuning secretly performs ppo," in *Second Workshop on Test-Time Adaptation: Putting Updates to the Test! at ICML 2025*.

[38] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70–90, 2018.

[39] K. A. Steen, P. Christiansen, H. Karstoft, and R. N. Jørgensen, "Using deep learning to challenge safety standard for highly autonomous machines in agriculture," *Journal of Imaging*, vol. 2, no. 1, p. 6, 2016.

[40] P. Christiansen, L. N. Nielsen, K. A. Steen, R. N. Jørgensen, and H. Karstoft, "Deepanomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field," *Sensors*, vol. 16, no. 11, p. 1904, 2016.

[41] M. Rahnemoonfar and C. Sheppard, "Deep count: fruit counting based on deep simulated learning," *Sensors*, vol. 17, no. 4, p. 905, 2017.

[42] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *sensors*, vol. 16, no. 8, p. 1222, 2016.

[43] K. Kuwata and R. Shibasaki, "Estimating crop yields with deep learning and remotely sensed data," in *2015 IEEE international geoscience and remote sensing symposium (IGARSS)*. IEEE, 2015, pp. 858–861.

[44] X. Song, G. Zhang, F. Liu, D. Li, Y. Zhao, and J. Yang, "Modeling spatio-temporal distribution of soil moisture by deep learning-based cellular automata model," *Journal of Arid Land*, vol. 8, no. 5, pp. 734–748, 2016.

[45] G. Sehgal, B. Gupta, K. Paneri, K. Singh, G. Sharma, and G. Shroff, "Crop planning using stochastic visual optimization," in *2017 IEEE Visualization in Data Science (VDS)*. IEEE, 2017, pp. 47–51.

[46] G. Han, D. K. P. Asiedu, and K. E. Bennin, "Plant disease detection with generative adversarial networks," *Heliyon*, vol. 11, no. 7, 2025.

[47] Y. Lu, D. Chen, E. Olaniyi, and Y. Huang, "Generative adversarial networks

(gans) for image augmentation in agriculture: A systematic review," *Computers and Electronics in Agriculture*, vol. 200, p. 107208, 2022.

[48] Z. ur Rahman, M. S. M. Asaari, H. Ibrahim, I. S. Z. Abidin, and M. K. Ishak, "Generative adversarial networks (gans) for image augmentation in farming: A review," *IEEE Access*, 2024.

[49] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[50] S. J. Prince, *Understanding deep learning*. MIT press, 2023.

[51] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," Tech. Rep., 1985.

[52] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096–1103.

[53] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[54] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations (ICLR)*, 2021.

[55] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[56] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, 2015, pp. 234–241.

[57] A. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International Conference on Machine Learning (ICML)*. PMLR, 2021, pp. 8162–8171.

[58] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proceedings of the 31st International Conference on Machine Learning (ICML)*. PMLR, 2014, pp. 1278–1286.

[59] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[61] OpenAI, "Introducing chatgpt," https://openai.com/blog/chatgpt, 2022, accessed: 2025-09-26.

[62] M. A. Beck, C.-Y. Liu, C. P. Bidinosti, C. J. Henry, C. M. Godee, and M. Ajmani, "An embedded system for the automated generation of labeled plant images to

enable machine learning applications in agriculture," *Plos one*, vol. 15, no. 12, p. e0243923, 2020.

[63] TerraByte Research Group, "Terrabyte: Agricultural imaging and ai research," https://terrabyte.acs.uwinnipeg.ca/, accessed: 2025-05-07.

[64] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," *arXiv preprint arXiv:2301.12597*, 2023.

[65] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *arXiv preprint arXiv:2304.08485*, 2023.

[66] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *Advances in neural information processing systems*, vol. 35, pp. 25 278–25 294, 2022.

[67] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[68] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29, 2016.

[69] M. Kang, J. Lee, G. Park, D. Ko, M. Kim, J. Kim, J. Park, S. Kim, J.-W. Ha, and K. Cho, "Scaling up gans for text-to-image synthesis," in *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 10 024–10 034.

[70] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in plant science*, vol. 7, p. 215232, 2016.

[71] P. K. Mensah, V. Akoto-Adjepong, K. Adu, M. A. Ayidzoe, E. A. Bediako, O. Nyarko-Boateng, S. Boateng, E. F. Donkor, F. U. Bawah, N. S. Awarayi *et al.*, "Ccmt: Dataset for crop pest and disease detection," *Data in Brief*, vol. 49, p. 109306, 2023.

[72] M. A. Gehan, N. Fahlgren, A. Abbasi, J. C. Berry, S. T. Callen, L. Chavez, A. N. Doust, M. J. Feldman, K. B. Gilbert, J. G. Hodge *et al.*, "Plantcv v2: Image analysis software for high-throughput plant phenotyping," *PeerJ*, vol. 5, p. e4088, 2017.

[73] G. Jocher, A. Chaurasia, and J. Qiu, "Yolo by ultralytics," Computer software, 2023, version 8.0.0. [Online]. Available: https://github.com/ultralytics/ultralytics

[74] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024.

[75] B. Gunel, J. Du, A. Conneau, and V. Stoyanov, "Supervised contrastive learning for pre-trained language model fine-tuning," *arXiv preprint arXiv:2011.01403*, 2020.

[76] G. Dong, H. Yuan, K. Lu, C. Li, M. Xue, D. Liu, W. Wang, Z. Yuan, C. Zhou,

and J. Zhou, "How abilities in large language models are affected by supervised fine-tuning data composition," *arXiv preprint arXiv:2310.05492*, 2023.

[77] A. Huang, A. Block, Q. Liu, N. Jiang, A. Krishnamurthy, and D. J. Foster, "Is best-of-n the best of them? coverage, scaling, and optimality in inference-time alignment," *arXiv preprint arXiv:2503.21878*, 2025.

[78] Y. Chow, G. Tennenholtz, I. Gur, V. Zhuang, B. Dai, S. Thiagarajan, C. Boutilier, R. Agarwal, A. Kumar, and A. Faust, "Inference-aware fine-tuning for best-of-n sampling in large language models," *arXiv preprint arXiv:2412.15287*, 2024.