



Deep Learning Project

# “Image Outpainting” using Residual Skip Connections



Anshika Singh



Harsh Mishra



Harsh Siroya



Rohit Singh



Vaisakh



Ronak Mir



# ***Problem Statement***



- In agricultural datasets, crop images are often incomplete, with missing or obscured regions. These gaps hinder their use in critical tasks like disease prediction, weed detection, and accurate visual analysis.
- Existing methods like image inpainting address missing areas within an image but are not optimized for extending beyond its limits, especially in domain-specific contexts like agriculture.



# What is Image Outpainting ?

Image outpainting or extrapolation is the technique of generating an image beyond its boundaries (given a certain portion of the image, we make a complete image- generating what is missing from a small portion).

## Project Objectives

- The goal of this project is to develop a deep learning model capable of performing image outpainting on a crop (specifically rice) dataset.
- The model will leverage advanced techniques such as hybrid residual skip connections .
- The outpainting model will be designed to generate realistic and contextually coherent extensions of the original crop images, overcoming the constraints imposed by limited image data.



# Literature Reviewed

## Image Outpainting and Harmonization using GAN (2020)

### Architecture:

#### Generator Network (G):

- Based on inpainting architecture.
- Uses context encoder with 6 convolutional layers to downsample input.
- Decoder upsamples using deconvolutional layers to restore original image size.

#### Discriminator Network (D):

- Estimates probability of image being real or generated.
- Operates on the entire outpainted image to avoid inconsistencies.
- Outputs a 24x24 grid of probabilities, with errors averaged during training.

**MSE:** 0.0230

### Image Outpainting and Harmonization using Generative Adversarial Networks

Basile Van Hoorick

Columbia University

basile.vanhoorick@columbia.edu

**Abstract** Although the inherently ambiguous task of predicting what resides beyond all four edges of an image has rarely been explored before, we demonstrate that GANs hold powerful potential in producing reasonable extrapolations. Two outpainting methods are proposed that aim to instigate this line of research: the first approach uses a context encoder inspired by common inpainting architectures and paradigms, while the second approach adds an extra post-processing step using a single-image generative model. This way, the hallucinated details are integrated with the style of the original image, in an attempt to further boost the quality of the result and possibly allow for arbitrary output resolutions to be supported.

#### I. INTRODUCTION

When presented with an incomplete image, humans are excellent at filling in the blanks and producing a realistic explanation for what could be missing. Image inpainting is a well-studied problem that replicates this behaviour, often tasking deep neural networks with trying to understand the semantic content of natural images in order to recover the missing regions of a photo. However, the spatially inverted variant of this problem is even more challenging and, with a small play on words, can be denoted *outpainting*. The problem statement is shown in Figure 1, essentially, the task is to extrapolate the image content rather than to interpolate within an image. More

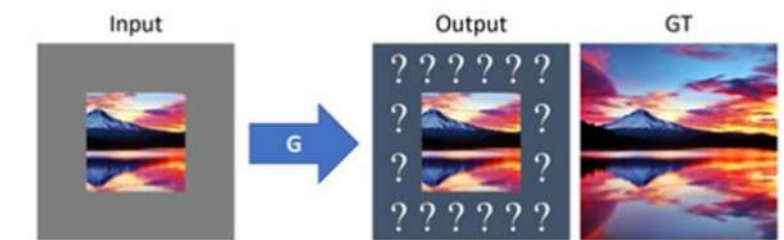


FIG. 1: Image outpainting idea.



[cs.CV] 15 Feb 2020



# Literature Reviewed

## Pathak et al.'s Context Encoder (2016)

- Introduced adversarial training with autoencoders for image inpainting, forecasting missing image segments based on surrounding context.
- A foundational work that paved the way for tasks like image outpainting and realistic completions in deep learning.

## Lizuka et al.'s Globally and Locally Consistent Image Completion (2017)

- Proposed a novel GAN architecture with dual discriminators:
  - Local discriminator for fine texture details.
  - Global discriminator for overall image coherence.
- Revolutionized visually consistent image completion by ensuring the extensions blend realistically with the background.




Fig. 1. Image completion results by our approach. The masked area is shown in white. Our approach can generate novel fragments that are not present elsewhere in the image, such as needed for completing faces; this is not possible with patch-based methods.

We present a novel approach for image completion that results in images that are both locally and globally consistent. With a fully-convolutional neural network, we can complete images of arbitrary resolutions by filling-in missing regions of any shape. To train this image completion network to be consistent, we use global and local context discriminators that are trained to distinguish real images from completed ones. The global discriminator looks at the entire image to assess if it is coherent as a whole, while the local discriminator looks only at a small area centered at the completed region to ensure the local consistency of the generated patches. The image completion network is then trained to fool the both context discriminator networks, which requires it to generate images that are indistinguishable from real ones with regard to overall consistency as well as in details. We show that our approach can be used to complete a wide variety of scenes. Furthermore, in contrast with the patch-based approaches such as PatchMatch, our approach can generate fragments that do not appear elsewhere in the image, which allows us to naturally complete the images of objects with familiar and highly specific structures, such as faces.

CCS Concepts: • **Computing methodologies** → **Image processing**; *Neural networks*;

Additional Key Words and Phrases: image completion, convolutional neural network

This work was partially supported by JST ACT-I Grant Number JPMJPR16U3 and JST CREST Grant Number JPMJCR14D1.  
© 2017 Copyright held by the owner/author(s). This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/http://dx.doi.org/10.1145/3072959.3073659>.

### ACM Reference format:

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and Locally Consistent Image Completion. *ACM Trans. Graph.* 36, 4, Article 107 (July 2017), 13 pages.  
DOI: <http://dx.doi.org/10.1145/3072959.3073659>

### 1 INTRODUCTION

Image completion is a technique that allows filling-in target regions with alternative contents. This allows removing unwanted objects or generating occluded regions for image-based 3D reconstruction. Although many approaches have been proposed for image completion, such as patch-based image synthesis [Barnes et al. 2009; Darabi et al. 2012; Huang et al. 2014; Simakov et al. 2008; Wexler et al. 2007], it remains a challenging problem because it often requires high-level recognition of scenes. Not only is it necessary to complete textured patterns, it is also important to understand the anatomy of the scene and objects being completed. Based on this observation, in this work we consider both the local continuity and the global composition of the scene, in a single framework for image completion.

We leverage a fully convolutional network as the basis of our approach, and propose a novel architecture that results in both locally and globally consistent natural image completion. Our architecture is composed of three networks: a completion network, a global context discriminator, and a local context discriminator. The completion network is fully convolutional and used to complete



# Dataset Description

- We use crop – **rice dataset** composed of diverse images captured in real agricultural environments.
- The data has been sourced from **Kaggle**
- The dataset was collected from different cities in Pakistan such as Kandhkot, Shikarpur, Sukkur, Moro, and Kashmore.
- This dataset provides visual data that can be used for agricultural tasks.
- The dataset is composed of **1007** images.
  1. Healthy crop images : 501
  2. Unhealthy crop images: 506





# ***Data Preprocessing***

The images had irregular dimensions; some were excessively long, while others were unusually wide.

- **The data is transformed using:**

- **Resize:** Resizes images to the target output size (e.g., 192x192).
- **Center\_Crop:** Crops the image to the desired size (e.g., 128x128).
- **Random\_Horizontal\_Flip:** Randomly flips the images horizontally as a data augmentation technique.
- **ToTensor:** Converts images to PyTorch tensors for model input.





# First Model Methodology



## Model Architecture

### Generator (CEGenerator):

- Encoder: Downsamples the input image through convolutional layers, capturing contextual information. We have 6 downsampling blocks
- Bottleneck: Reduces the feature map to a size of 4000 channels
- Decoder: Upsamples the feature map with transposed convolutions to reconstruct the image.

### Discriminator (CEDiscriminator):

- Evaluation: Uses 4 convolutional layers to downsample the image, learning higher-level features.
- Output: Outputs a scalar value indicating whether the image is real or fake, used for adversarial loss to improve the generator.



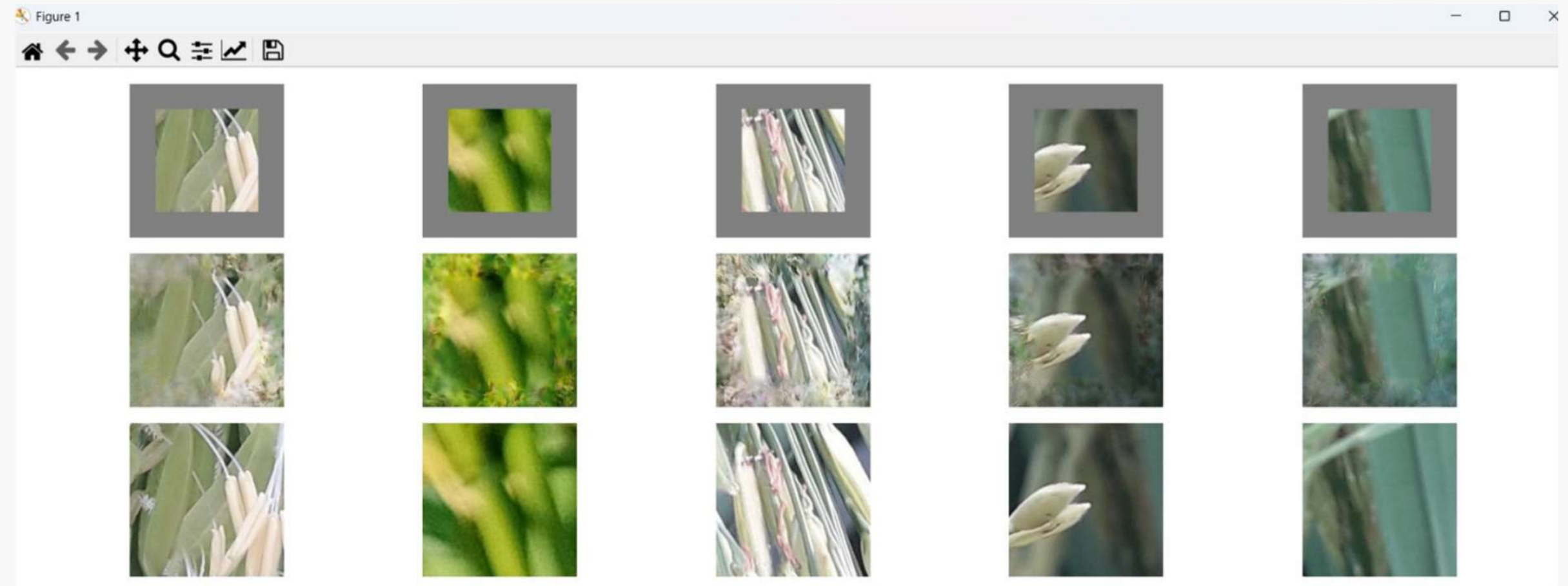
# Results



***Input Image***

***Output Image***

***Ground Truth***



- Evaluation Metrics: we used the MSE (Mean Squared Error) and the MAE ( Mean Absolute Error )

***The MSE for the first model was 0.0192***

***The MAE for the first model was 0.092***

```
. This limits the functions that could  
explicitly allowlisted by the user via  
t have full control of the loaded fi  
state_dict = torch.load(model_path  
Mean MSE: 0.0192  
Mean MAE: 0.0920  
PS C:\Users\govin\DeepLearning>
```



# Second Model Architecture

## Encoder:

- Convolutional Layers: Apply a series of convolutional layers (64 to 256 filters), progressively reducing spatial dimensions.
- Dilated Convolutions: Some layers use dilated convolutions to capture larger receptive fields and handle missing content.

## Residual and Skip Connections:

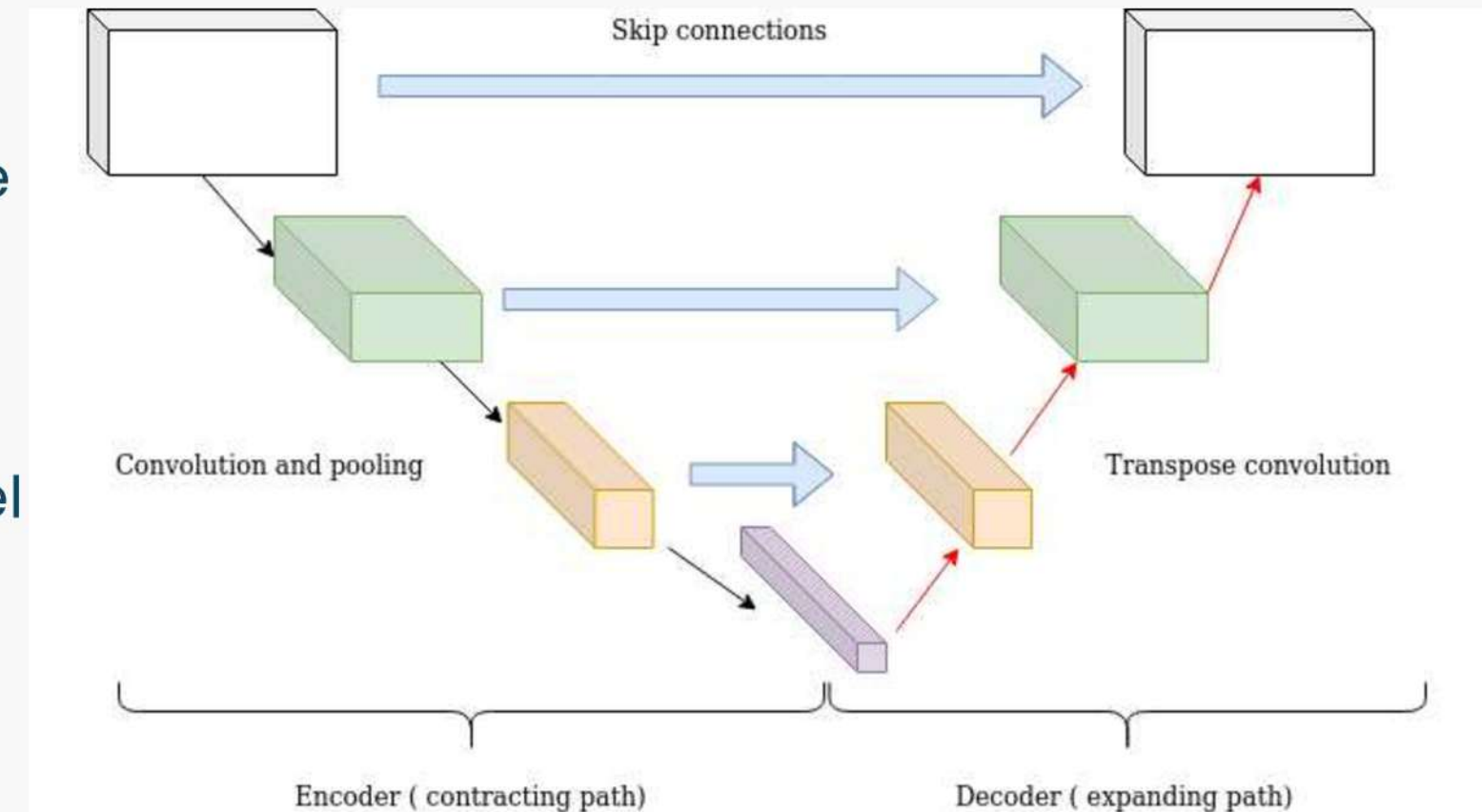
- Residual Connection: Passes features from intermediate encoder layers directly to the decoder, retaining fine-grained details.
- Skip Connections: Connect earlier encoder layers to corresponding decoder layers to help preserve low-level features during upsampling.

## Latent Space:

- Intermediate layers of the encoder form a latent space, capturing contextual information about the image, including missing regions.

## Decoder:

- Transposed Convolutions (Upsampling): Uses deconvolutions to progressively upsample the feature maps from the latent space back to the original image resolution.
- Sigmoid Activation: Applied to the final image to scale pixel values to the range  $[0, 1]$ .





# Second Model

## Generator Completion Network

| Layer              | Details  | Activation Function | Purpose                                     |
|--------------------|--|---------------------|---|
| Conv1              | Conv2D (4, 64, kernel=5, stride=1, padding=2)                        | ReLU                | Extracts low-level features from input      |
| Conv2              | Conv2D (64, 128, kernel=3, stride=2, padding=1)                      | ReLU                | Downsampling with increased depth           |
| Conv3, Conv4       | Conv2D (128→128, 256→256, kernel=3, stride=1/2, padding=1)           | ReLU                | Multi-scale feature extraction              |
| Conv5-Conv10       | Conv2D (256→256, kernel=3, dilation=2/4/8/16, padding=corresponding) | ReLU                | Captures large receptive fields for context |
| Conv11, Conv12     | Conv2D (256→256, kernel=3, stride=1, padding=1)                      | ReLU                | Further feature refinement                  |
| Deconv13, Deconv15 | ConvTranspose2D (256→128, 128→64, kernel=4, stride=2, padding=1)     | ReLU                | Upsampling layers to reconstruct image size |
| Conv14, Conv16     | Conv2D (128→128, 64→32, kernel=3, stride=1, padding=1)               | ReLU                | Refinement of upsampled features            |
| Conv17             | Conv2D (32→3, kernel=3, stride=1, padding=1)                         | Sigmoid             | Produces final 3-channel output             |

## Local Discriminator

| Layer       | Details                                       | Activation Function | Purpose  |
|-------------|---|---------------------|--|
| Conv1-Conv5 | Conv2D (C→512, kernel=5, stride=2, padding=2) | ReLU + BatchNorm    | Extracts multi-scale features from patches           |
| Flatten6    | Flatten                                       | None                | Flattens feature map for fully connected input       |
| Linear6     | Linear (Input→1024)                           | ReLU                | Produces latent representation for the discriminator |

## Global Discriminator

| Layer       | Details  | Activation Function | Purpose  |
|-------------|--|---------------------|--|
| Conv1-Conv5 | Conv2D (C→512, kernel=5, stride=2, padding=2)                  | ReLU + BatchNorm    | Global feature extraction from entire input          |
| Conv6       | Conv2D (512→512, kernel=5, stride=2, padding=2) (for places2 ) | ReLU + BatchNorm    | Additional global feature extraction                 |
| Flatten6/7  | Flatten  | None                | Flattens feature map for fully connected input       |
| Linear6/7   | Linear (Input→1024)  | ReLU                | Produces latent representation for the discriminator |



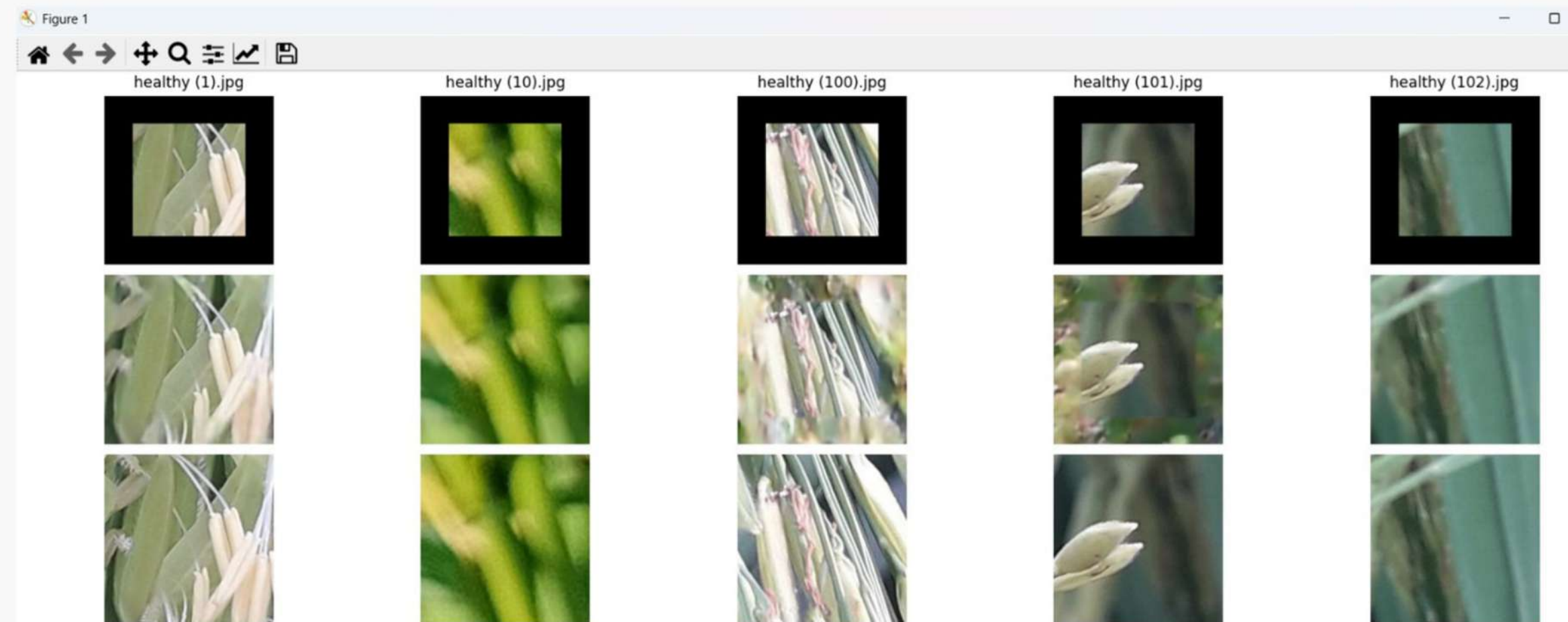
# Results



**Input Image**

**Output Image**

**Ground Truth**



- Evaluation Metrics: we used the MSE (Mean Squared Error) and the MAE ( Mean Absolute Error )

**The MSE for the first model was 0.016447**

**The MAE for the first model was 0.075631**

```
b.com/pytorch/pytorch/blob/main/SECURITY.md#u
rue`. This limits the functions that could be
e explicitly allowlisted by the user via `tor
don't have full control of the loaded file. P
model.load_state_dict(torch.load(model_path
Calculating metrics: 100%|
Average MSE: 0.016447
Average MAE: 0.075631
PS C:\Users\gavin\Documents>
```



# Novelty

- Skip Connections: Added residual connections to enhance gradient flow and feature propagation by bypassing intermediate layers.
- Dilated Convolutions: Expanded the receptive field to capture larger contextual information without increasing parameters.
- Local and Global Discrimination: Ensured fine-grained details and overall image consistency using both local and global discriminators.

## Why ?

- Residual connection is introduced in the CompletionNetwork ( $x = x + \text{residual}$ ), allowing better gradient flow, improving training stability and model performance.
- The addition of skip connections helps retain spatial information by skipping layers, enhancing feature reuse and performance in deep architectures.



# Key learnings from Experimentation

## **1. Importance of Skip Connections in Feature Preservation:**

- Experimenting with the base model revealed that skip connections are critical for retaining both high- and low-level features during outpainting.
- This learning has driven the development of a U-Net-like structure to enhance the quality of generated images.

## **2. Role of Dense Residual Learning:**

- Dense residual blocks proved essential for preserving finer image details and ensuring smooth transitions in outpainted regions.
- The use of these blocks in our final model is expected to address the limitations of the base approach.

## **3. Training Stability Challenges:**

- The base model training highlighted the instability of GANs, particularly when generating complex outpainting tasks.
- This reinforced the need for progressive learning techniques to improve training reliability and output quality.

## **4. Insights into Model Scalability:**

- The base model provided an understanding of computational resource requirements, helping in planning for a more efficient implementation of the full model.



# *Conclusion*



---

In this project, we tackled the challenge of image outpainting for crop datasets, a relatively underexplored field in the context of agriculture.

Our GAN-based approach has demonstrated the ability to generate realistic extensions of incomplete crop images, thus enhancing the quality and usability of agricultural data.

By incorporating innovations like dense residual learning with skip connection, we were able to preserve fine details and improve the quality of outpainted regions.

---





# References



- Y. Kim, J. Lim, and C. Kim, "Painting Outside As Inside: Edge Guided Image Outpainting via Bidirectional Rearrangement with Progressive Step Learning," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2021, pp. 993–1002.
- C. Cheng, C. Lu, and X. Ren, "InOut: Diverse Image Outpainting via GAN Inversion," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 14090–14100.
- K. Wang, J. H. Pan, C. C. Loy, and L. Lin, "Diverse Image Completion with Bidirectional GANs," IEEE Transactions on Image Processing, vol. 29, pp. 3497–3506, 2020.
- M. Mirzaei and M. Abbasi, "A hybrid machine learning approach for intelligent IoT energy management systems in smart buildings," Neural Networks, vol. 156, pp. 225–233, 2023.
- S. Zor, B. Baykal, and O. Koncagul, "Deep neural networks in decision support systems: Application of multi-objective optimization in power allocation problem," Expert Systems with Applications, vol. 213, 2022.
- A. Liu and Y. Huang, "Generative Adversarial Networks for Image Completion," Stanford University, CS230 Project Report, Spring 2018







***Thank you***

