# Image Super Resolution

B.Laxmi Sreenivas (IMT2020510)
Rithvik Ramasani (IMT2020543)

## Introduction

Image super-resolution (ISR) is a computer vision technique designed to enhance the quality and resolution of images by converting low-resolution (LR) images into high-resolution (HR) versions. This process improves image details and sharpness, contributing to various real-world applications, such as medical imaging, satellite imaging, surveillance, security, and astronomical imaging.

Our implementation focused on creating a deep learning model for image super-resolution, utilising the WaveMixSR architecture. This model aims to boost image quality while remaining efficient in terms of computational resource use. The rationale behind this approach is the growing demand for high-quality images across industries, paired with the challenges of computational constraints.

In developing our WaveMixSR-based ISR model, we took a methodical approach, beginning with the exploration of various wavelet families. Our objective was to determine the best wavelet family to facilitate super-resolution. This phase was critical because different wavelets have varying impacts on image enhancement, and choosing the right one could make a significant difference in the final image quality.

Next, we evaluated several pre-trained networks for feature extraction, such as VGG11 and ResNet, to determine which provided the most reliable foundation for our super-resolution tasks. These pre-trained networks offer powerful feature extraction capabilities, essential for identifying intricate patterns and details within images.

With a robust WaveMixSR-based network in place, we proceeded to enhance the model with a discriminator, incorporating it into a Generative Adversarial Network (GAN) framework. This step involved adversarial training, a technique that encourages the generator (our WaveMixSR model) to create more realistic high-resolution images by constantly challenging it with a discriminator that evaluates the authenticity of the generated images.

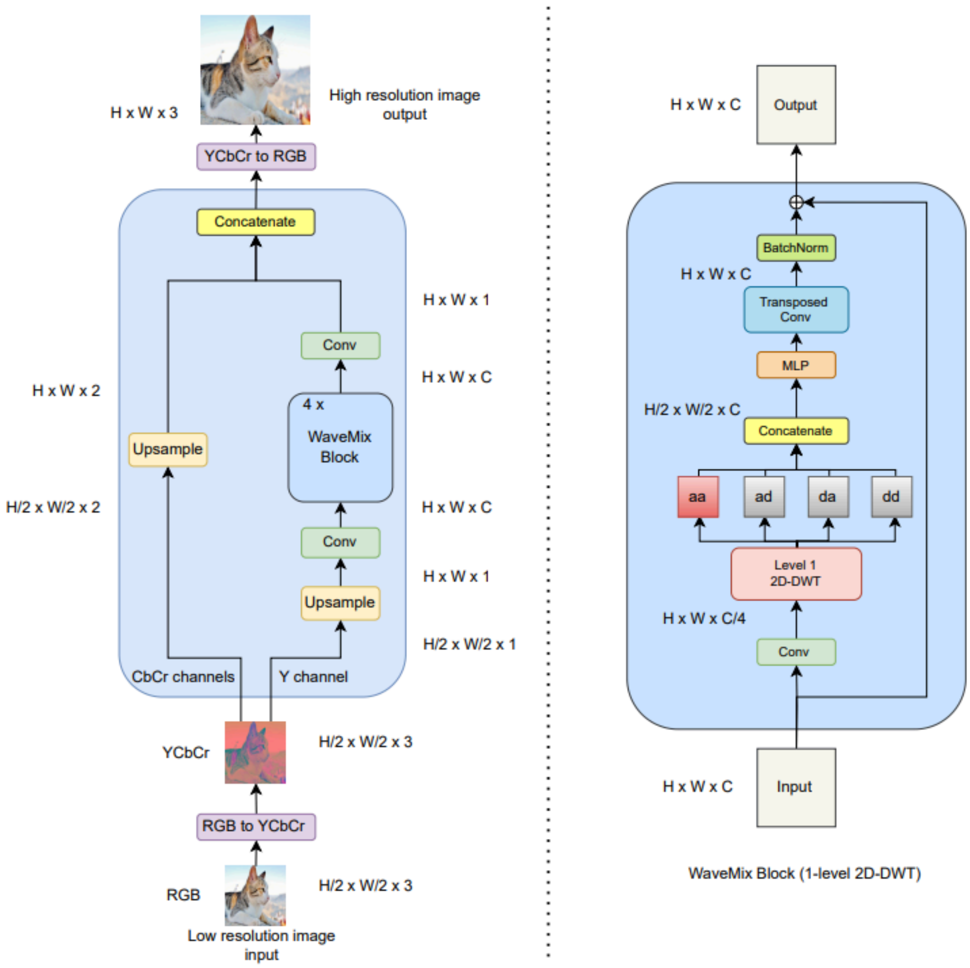
The resulting model combines the benefits of WaveMixSR, the optimised wavelet family, and the advanced feature extraction capabilities of pre-trained networks, with the added realism provided by GAN-based adversarial training. This comprehensive approach allows our ISR model to produce high-resolution images that are both visually appealing and computationally efficient, making it suitable for a wide range of applications.

# WaveMixSR Model

The WaveMixSR model is a novel approach to image super-resolution, combining the strengths of Convolutional Neural Networks (CNNs) and Vision Transformers (VTs). Its innovative architecture leverages spatial token-mixing through a 2D discrete wavelet transform (DWT), allowing efficient communication between different regions of an image. This unique combination contributes to improved computational efficiency and requires less training data compared to models based purely on VTs.

## *Key Features*

- 2D Discrete Wavelet Transform (DWT): The use of 2D DWT is central to the WaveMixSR model, enabling efficient spatial token-mixing and reducing computational demands. This approach allows the model to capture scale-invariant features, facilitating the reconstruction of high-resolution images.
- Dual-Path Processing: WaveMixSR employs a dual-path architecture to process the luminance (Y-channel) and chrominance (CbCr channels) separately.
    - *Y-Channel (Luminance):* This path is given priority because it contains higher information content, crucial for detail reconstruction.
    - *CbCr Channels (Chrominance):* These channels are responsible for color information and are upsampled directly, keeping the architecture computationally efficient.




## *WaveMix Blocks*

The core of the WaveMixSR model lies in its WaveMix blocks, particularly within the Y-channel path. These blocks consist of several key components, each contributing to the model's effectiveness in detail reconstruction and sharpening:

- Convolution: This step provides learnable feature extraction, helping the model identify significant patterns within the image.
- 2D DWT: Used for spatial token-mixing and to introduce scale-invariant features, aiding in the reconstruction of detailed high-resolution images.
- Multi-Layer Perceptron (MLP): This component facilitates channel-wise feature mixing, enabling the model to process information from different channels effectively.
- Transposed Convolution: This step is used to restore spatial resolution, helping to produce high-resolution images from the features extracted and mixed in previous layers.
- Residual Connections: These connections are employed to improve gradient flow, mitigating issues related to deep networks, such as vanishing gradients.

### *Upsampling*

To upscale the images to their final high-resolution form, WaveMixSR uses efficient non-parametric methods like bilinear or bicubic interpolation. These methods offer a computationally efficient way to upsample images without adding excessive complexity to the model.

Overall, the WaveMixSR model's design, with its focus on computational efficiency, effective detail reconstruction, and dual-path architecture, provides a robust foundation for image super-resolution. It integrates the best aspects of CNNs and VTs while incorporating innovative features like 2D DWT and WaveMix blocks to achieve high-quality results.

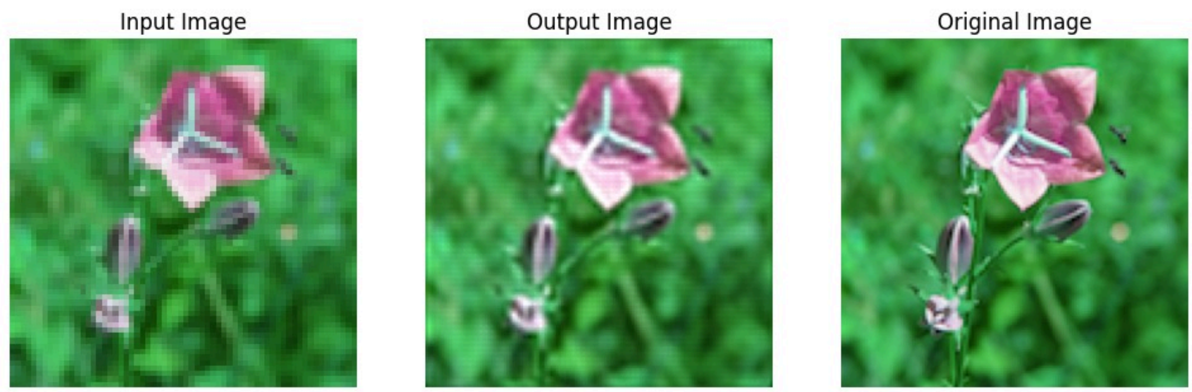
# Wavelet Family Analysis

Epochs = 50

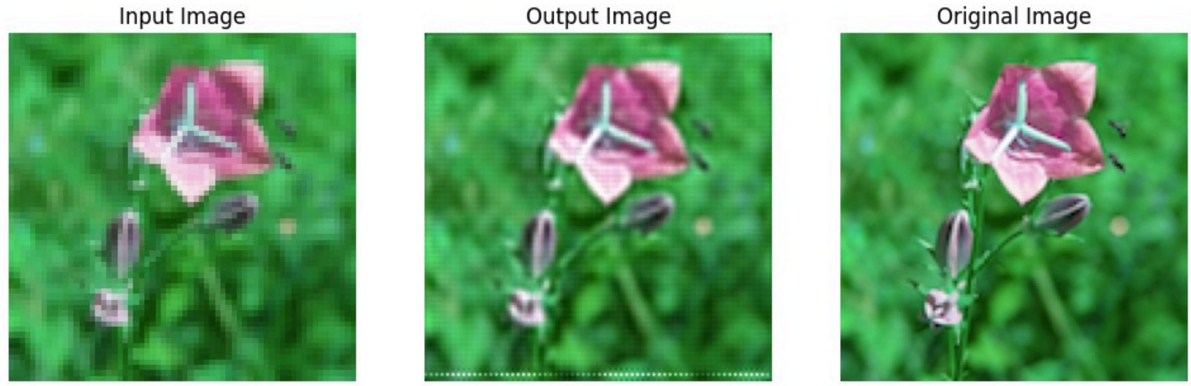
| Wavelet Family Name | PSNR Loss | SSIM Loss |
| --- | --- | --- |
| Haar Wavelets | 29.1 | 0.87 |
| DB1 Wavelet | 28.45 | 0.79 |
| Symlet Wavelet | 28.7 | 0.81 |

### *Results of various wavelets*

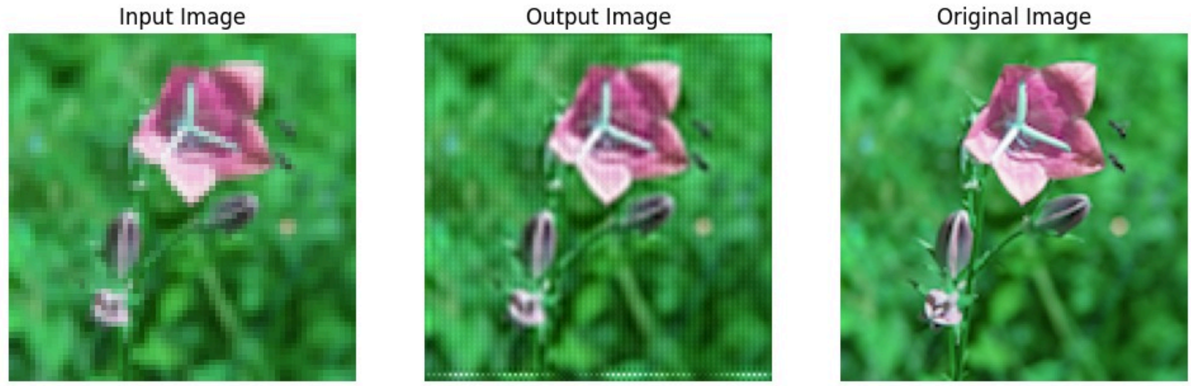
Haar Wavelets



DB1 Wavelets




Symlet Wavelet




After experimenting with three distinct wavelet families, we found that the Haar wavelet delivered superior results compared to the other two. DB and Symlet wavelets are causing noise at the borders of the images. As a result, we decided to continue with the Haar wavelet in our WaveMixSR model.

# Pre-Trained Network Integration

Pre-trained networks like VGG11 and ResNet18 have demonstrated remarkable feature extraction capabilities in various image processing tasks, making them valuable tools for image super-resolution models like WaveMixSR. By integrating parts of these pre-trained networks, we leverage their ability to extract meaningful patterns and features from images, enhancing the detail reconstruction process in super-resolution tasks.

*<u>VGG11:</u>*

The layers selected are:

- Convolutional Layer 1:
  - Input size: (224, 224, 3)
  - Output size: (224, 224, 64)
  - The convolution is followed by a ReLU (Rectified Linear Unit) activation function.
- Max Pooling Layer 1:
  - Input size: (224, 224, 64)
  - Output size: (112, 112, 64)
- Convolutional Layer 2:
  - Input size: (112, 112, 64)
  - Output size: (112, 112, 128)
  - This is again followed by a ReLU activation function.

*<u>ResNet18:</u>*

The layers selected are:

- Convolutional Layer 1 (conv1):
  - Input size: (224, 224, 3)
  - Output size: (112, 112, 64)
  - The convolution is followed by a batch normalisation layer (bn1) to stabilise the training process and a ReLU (Rectified Linear Unit) activation function.
- Max Pooling Layer 1:
  - Input size: (112, 112, 64)
  - Output size: (56, 56, 64)
  - This layer applies a 3x3 max pooling operation with a stride of 2, which reduces the spatial dimensions and retains the most significant features.
- Residual Block Layer 1:
  - Input size: (56, 56, 64)
  - Output size: (56, 56, 64)
  - This layer consists of two convolutional layers, each with a 3x3 kernel, followed by batch normalisation and ReLU activation. The residual connection allows the model to learn more complex features while minimising the risk of vanishing gradients.
- Residual Block Layer 2:
  - Input size: (56, 56, 64)

- Output size: (28, 28, 128)
- This layer doubles the number of output channels to 128, indicating the progression to more complex feature extraction. It has two 3x3 convolutional layers, each followed by batch normalisation and ReLU activation. The residual connection ensures that the information from earlier layers is preserved, enhancing gradient flow through the network.

Loss Considered is PSNR Loss

| Network | Epochs=10 | Epochs=20 |
|---|---|---|
| VGG11 | 30.1 | 31.1 |
| ResNet18 | 31.8 | 32.5 |

## *Results of Pre-Trained Networks*

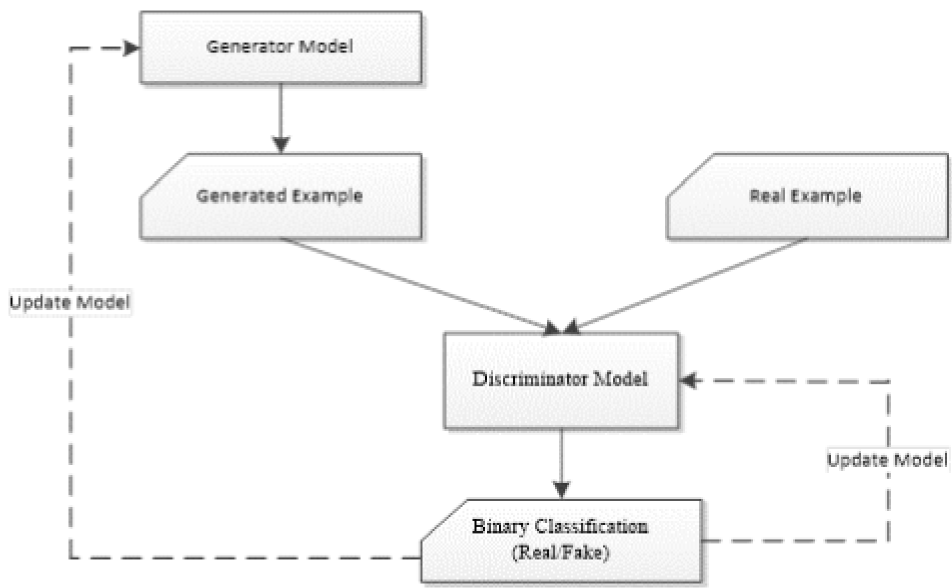
ResNet18


VGG11

The results from the ResNet18 model were more similar to the original image, leading us to consider it a better model for our purposes. We decided to proceed with the ResNet18. Furthermore, using a pre-trained network yielded better images compared to the base model, which produced much grainier results without pre-training.

ResNet18 performing better than VGG11 is expected as the state-of-the-art model EDSR, also employs residual learning like ResNet18. Better results of the pre-trained network compared to the base model can also be attributed to the additional focus given to CbCr channels, which are initially just upsampled without following the feature extraction path.

# Adversarial Training




Generative Adversarial Networks (GANs) are a class of deep learning models that consist of two neural networks—a generator and a discriminator—that are trained simultaneously in an adversarial manner.

## *How GANs Work?*

- Generator: The generator's task is to create realistic-looking data samples, such as images, from random noise or other input data. Its goal is to "fool" the discriminator into believing that its outputs are real.
- Discriminator: The discriminator is trained to distinguish between real data samples and those generated by the generator. It provides feedback to the generator on how well it is performing by indicating whether it successfully "fooled" the discriminator.
- Adversarial Training: The generator and discriminator are trained simultaneously but with opposing goals—the generator aims to create realistic outputs, while the discriminator aims to correctly classify them as real or fake. The training continues until the generator produces outputs that the discriminator cannot reliably distinguish from real data.

## *Why Use GANs in Image Super-Resolution?*

GANs are particularly useful in image super-resolution for several reasons:

- Improved Realism: GANs can generate high-resolution images with realistic textures and details, which is often challenging with traditional super-resolution methods. By competing against the discriminator, the generator learns to create images with intricate patterns and features.
- Enhanced Detail Reconstruction: In adversarial training, the discriminator encourages the generator to create outputs with high perceptual quality. This leads to super-resolution results that are more visually appealing and closer to natural images.
- Robust Training: The adversarial setup ensures continuous feedback between the generator and discriminator, promoting a more robust learning process. This dynamic can improve the generator's ability to capture complex features.

## *Discriminator & Generator Used*

The discriminator used for the model is described below:

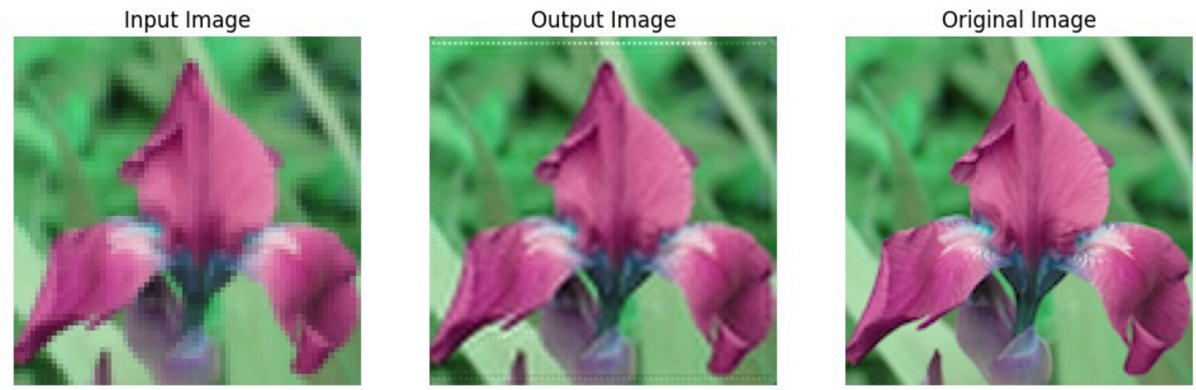
```
x = F.leaky_relu(self.conv1(x))
x = F.leaky_relu(self.bn2(self.conv2(x)))
x = x.view(-1, 128 * 8 * 8)
x = F.leaky_relu(self.fc1(x))
x = self.dropout(x)
x = F.leaky_relu(self.fc2(x))
```

```
x = self.dropout(x)
x = self.fc3(x)
x = self.fc4(x)
x = self.fc5(x)
x = self.fc6(x)
x = self.fc7(x)
x = torch.sigmoid(x)
```

The generator network used was WaveMixSR with ResNet18 Backbone, which has given us the best results so far.

*Results of Adervsarial Training*

Epochs = 20
PSNR Loss = 34.46
Discriminator True Cross Entropy Loss = 0.0011899899538150589
Discriminator False Cross Entropy Loss = 0.0004164262071752967




This model has given us the best results, which are both visually and loss-wise better than any of the previous models.

# References

1. WaveMixSR: Resource-efficient Neural Network for Image Super-resolution.
2. Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution, 2017.
3. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network [Link]