**REPORT\_PART1: LAB2**

**Pair Programming Group 19 - Divya Neelamegam, Harika Boyina**

**Introduction:**

The advancement of deep learning in image processing has paved the way for innovative approaches to enhance image resolution. Super-resolution techniques, particularly those employing Generative Adversarial Networks (GANs), have shown promising results in improving the quality and details of up-scaled images. This report details the deployment of SRGAN, highlighting its architecture, training process, and the results obtained from testing on low-resolution images.

**Model Architecture**

The SRGAN model comprises two primary components:

**Generator:** Responsible for producing high-resolution images from low-resolution inputs. The generator architecture is built upon a ResNet model that includes multiple residual blocks with convolutional layers, batch normalization, and ParametricReLU activation to enhance feature extraction without adding computational complexity.

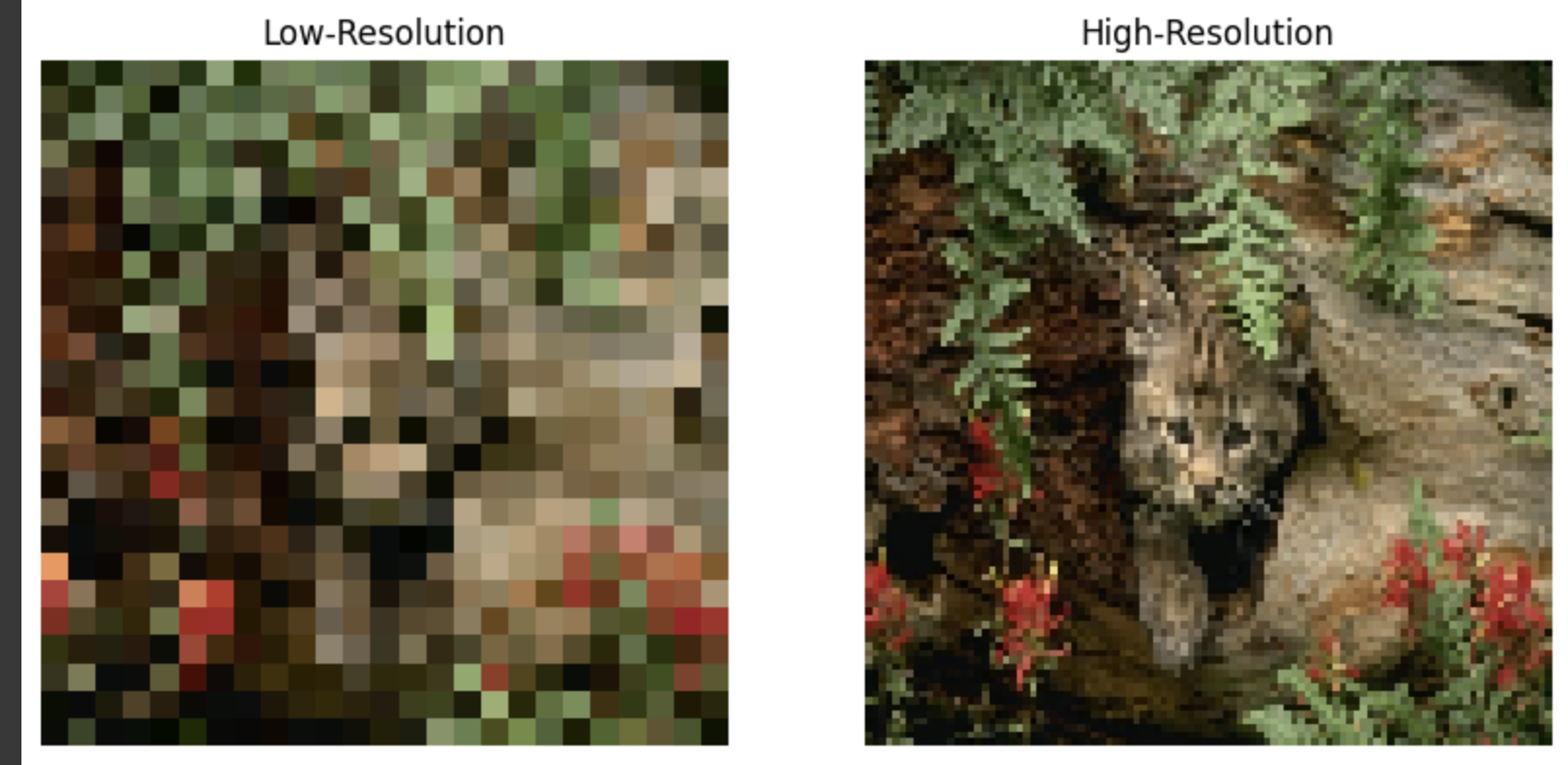
**Discriminator**: Acts as a critic that distinguishes between high-resolution images generated by the Generator and the authentic high-resolution images. It consists of a deep convolutional network which progressively downsamples its input.

**Feature Extraction:** A pre-trained VGG19 model, truncated at the 18th layer, serves as a feature extractor to calculate content loss. This method ensures that the generator not only fools the discriminator but also aligns its output closely with the feature-rich aspects of the target high-resolution images.

**Training Process**

**Data Preparation:** Images are sourced and stored in high-resolution and low-resolution directories. During preprocessing, images undergo transformations such as normalization and conversion to tensors to prepare them for processing by the neural network.

Sample input image pair (LR and HR) :



**Loss Functions:**

Adversarial Loss: Employs Mean Squared Error (MSE) to measure the ability of the generator to fool the discriminator.

Content Loss: Utilizes L1 loss between features extracted from the generated images and the features of real images, emphasizing similarity in content beyond mere pixel values.

**Optimization**: Separate Adam optimizers with learning rates of 0.0002 and momentum parameters betas=(0.5, 0.999) are used for the generator and discriminator to stabilize training and promote convergence.

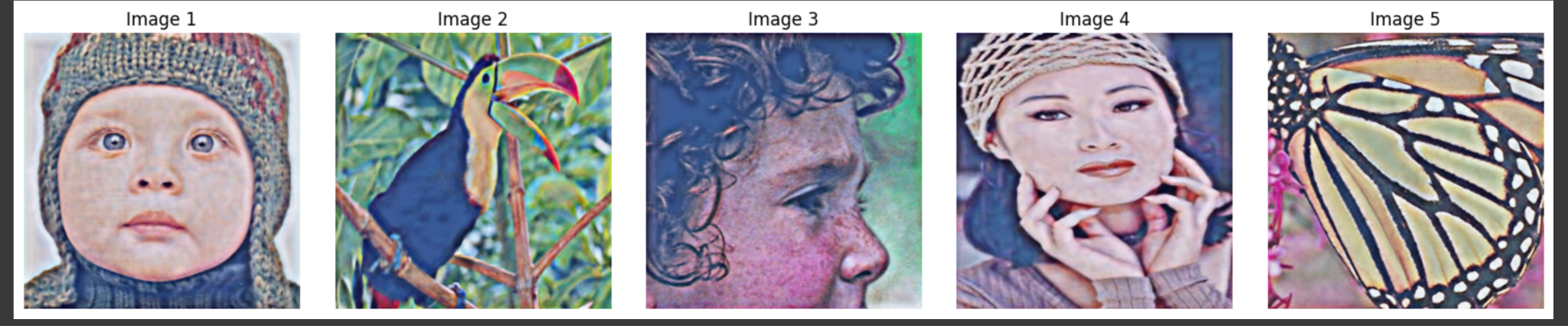
**Results**

After training the model with 250 epochs, the model performs well in generating SR images. The saved model is attached in this submission as **“srgan\_checkpoint.pth”**

**Evaluation**

The effectiveness of SRGAN is assessed through comparisons between the generated high-resolution images and their corresponding ground truths. Images are displayed using matplotlib to illustrate improvements in resolution and detail.

Test image generated:



**Challenges and Improvements**

* Training GANs, especially on high-resolution images, is computationally intensive. Optimizations such as model quantization were necessary to manage resource usage effectively.
* When the model is not defined properly, only blur images were generated with decreasing Discriminator loss and increasing generator loss.
* The model could perform even better if trained for more epochs for perfect output.

**Conclusion**

The implementation of SRGAN has demonstrated significant potential in enhancing low-resolution images to high-resolution. The model successfully learns to reconstruct high-quality images that are both visually appealing and close in detail to the original high-resolution images. Further research into more efficient training methods and advanced GAN architectures could lead to even better performance and broader application possibilities.