**Image Processing - COMP 6771 (Winter 2024)**

Computer Science and Software Engineering

Team Project Phase - 1

Submitted To **Dr. Yaser Esmaeili Salehani**

**Title:** **Photo-realistic Single Image Super-Resolution Using a Generative Adversarial Network**

**Submitted By: Group 15**

**Group members:**

|  |  |
| --- | --- |
| Chidambar Joshi | 40294870 |
| Sourav | 40228888 |
| Sindoora | 40199155 |
| Roshini Chukkapalli | 40198936 |

**INTRODUCTION**

SRGAN (Super Resolution Generative Adversarial Network) is an approach in the realm of image super- resolution that leverages the capabilities of Generative Adversarial Networks to enhance low resolution images to high resolution images. Unlike other image resolution techniques, SRGAN mainly focuses on the improvement of the perceptual quality of upscaled images. It utilises a dual component architecture with a generator that upscales the images and a discriminator whose function is to distinguish between the upscaled images and authentic high-resolution images. This method not only improves the resolution of the images but also facilitates the preservation of intricate details and textures, resulting in photo realistic super-resolved images.

**DATA DESCRIPTION**

We utilise the DIV2K dataset for evaluating the performance of our image processing algorithms. Below is a detailed description of the dataset:

**DIV2K Dataset**: DIV2K is a high-quality image dataset specifically designed for the task of image super-resolution and other related image processing challenges. It consists of 1,000 diverse, high-resolution images (2K resolution) that are divided into three subsets: a training set with 800 images, a validation set with 100 images, and a test set with another 100 images. The images in the DIV2K dataset cover a wide range of scenes, including natural landscapes, urban environments, and intricate textures, making it a comprehensive benchmark for evaluating super-resolution algorithms.

The DIV2K dataset is renowned for its high-quality images and diversity, providing a robust platform for developing and testing advanced image processing techniques. By using this dataset, researchers and practitioners can assess the effectiveness of their algorithms in enhancing image resolution while maintaining or improving image fidelity. The results obtained on the DIV2K dataset are often considered as a standard benchmark in the field of image super-resolution.

In our research, the DIV2K dataset serves as a crucial tool for validating the performance of our proposed image processing algorithms. By comparing the results on this dataset with those obtained from other benchmarks, we can ensure the competitiveness and generalizability of our methodologies in various real-world scenarios.

**METHODOLOGY AND APPLICATION**

The main methodology of SRGAN comprises of training the model with pairs of Low Resolution(LR) and High Resolution(HR) images. These image pairs are very important for the model to learn the transformation of Low Resolution images to High Resolution images. These LR images act as the input to the Generator which then attempts to upscale the images to match the resolution of corresponding HR images. These HR images act as the target to the Generator, enabling it to learn and mimic the high resolution details during the training process.

Architecture:

Super Resolution Generative Adversarial Network is a deep learning framework consisting of two major components.

* Generator: The functionality of Generator is to take low resolution images and generate a high resolution image. It is typically a deep convolutional network that learns to upgrade images by adding realistic details during the training process.
* Discrimator: The functionality of discriminator is to distinguish the high resolution image generated by the generator and the high resolution input image from the dataset. This is also a deep convolutional network trained to perform the task effectively.

Implementation:

In the project, the SRGAN is implemented using PyTorch library. The model’s architecture can be found in srgan\_model.py file which basically outlines the construction of generator and discriminator.

The construction of Generator employs a series of residual blocks for deep feature extraction and learning the intricate patterns required for upscaling. Upsampling blocks are used to increase the resolution of the images. The Generator’s main aim is to minimise the perceptual loss which measures the difference in feature representations of the generated image and target image computed by a pre-trained network.

The discriminator is designed in such a way that it classifies the real and generated images. This utilises a convolutional network with increasing depth to extract the features from an image, then proceeds by fully connected layers that outputs the probability of an image being real.

During the training process, both the Generator and discriminator are trained alternatively. The generator is trained to decrease the combination of content loss and adversarial loss, encouraging it to produce the images similar to that of target high resolution images so that the discriminator can no longer distinguish them. Similarly, the discriminator is trained to increase its ability to correctly classify the images. This training process is iterated until Generator produces high resolution images and the discriminator can no longer differentiate them.

Model Structure:

Generator:

Input Features: 3 (RGB channels)

Number of features: 64

Kernel Size: 3

Number of residual blocks: 16

Activation layer used: PReLU( Parametric ReLU) is used for intermediate layers and Tanh is used for output layers.

Upsampling: Uses two upsampler blocks for a scale of 4, each consisting of convolutions that upscale the image.

Initial and Final Convolution: 9\*9 convolution in the beginning and 3\*3 in the end.

Discrimator:

Input Features: 3 (RGB Channels)

Number of features: starts with 64 and doubles after each block.

Kernel Size: 3

Number of blocks: 3, each designed to half the feature map’s size.

Activation layer used: Leaky ReLU

Final Layer: A linear layer which is followed by a sigmoid function to classify images real or fake.

Training parameters and loss function:

Batch Size: 16

L2 Coefficient: 1.0

Adversarial Coefficient: 1e-3

Total Variation (TV) Loss Coefficient: 0.0

Pre-Training Epochs: 1000

Fine-Tuning Epochs: 4000

Scale: 4 (for upscaling)

Patch Size: 24

Feature Layer for Perceptual Loss: 'relu5\_4' from VGG19

Perceptual loss: Uses a pre-trained VGG19 model, targeting the relu5\_4 layer to calculate the mean squared error between the features of the high resolution and super resolved images.

**RESULTS**

Upon running the algorithm, it can be observed that the SRGAN model significantly improves the resolution of the provided images, making them look more sharper and detail- oriented.