Image Super-Resolution via Convolutional Neural Network:

1. Introduction

Low-resolution images often lack critical visual details, which can limit their utility in domains such as medical imaging, remote sensing, surveillance, and multimedia applications. Image super-resolution techniques aim to counteract the effects of faulty imaging hardware or software degradation by intelligently reconstructing high-resolution (HR) images from their low-resolution (LR) counterparts. This project employs the Super-Resolution Convolutional Neural Network (SRCNN) to reconstruct high-resolution images from low-resolution inputs, thereby enhancing image quality and preserving important features.

2. Model Architecture

The SRCNN (Super-Resolution Convolutional Neural Network) is a three-layer convolutional network designed to enhance low-resolution images. Its architecture can be broken down into three conceptual steps:

- **Layer 1: Patch Extraction and Representation:** Extracts overlapping image patches and represents them in a high-dimensional feature space.
- **Layer 2: Non-linear Mapping:** Performs non-linear mapping to learn complex relationships between low- and high-resolution patches.
- **Layer 3: Reconstruction:** Reconstructs the final high-resolution image from the learned features.

Activation: ReLU is used in the first two layers to introduce non-linearity, enabling the network to capture intricate image details.

3. Dataset and Preprocessing

The project utilizes two standard datasets for training and evaluation:

- **Training Dataset (91-image):** A collection of 91 high-quality images, preprocessed into smaller patches to train the SRCNN model for super-resolution tasks. It provides diverse textures and structures to improve model generalization.
- **Evaluation Dataset (Set5):** A benchmark dataset consisting of 5 widely used images to evaluate super-resolution performance. It allows quantitative assessment using metrics like PSNR and SSIM.

4. Experimental Setup

The model's performance was evaluated by varying the upscaling factor and the learning rate. The primary evaluation metric was the **Peak Signal-to-Noise Ratio (PSNR)**.

- **Hyperparameters:**
  - **Scale Factors:** 2, 3, and 4.
  - **Learning Rates (for Scale=3):** 1e-3, 1e-4, 1e-5.
  - **Batch Size:** 16.
  - **Epochs:** 100.

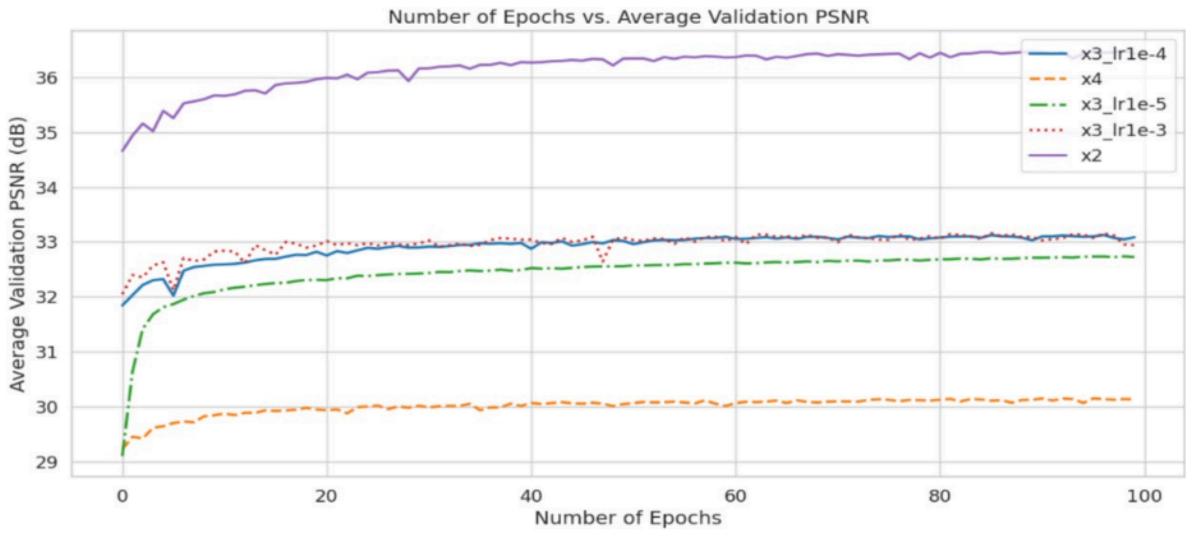
These experiments investigate how different scales and learning rates influence the network's validation performance.

5. Results and Analysis

5.1. Quantitative Results

The table below summarizes the best performance achieved for each experimental configuration.

Model	Best PSNR	Best Epoch	Final Loss
x2	36.4782	98	0.000543483
x3_lr1e-3	33.1602	85	0.00113819
x3_lr1e-4	33.1287	96	0.00116326
x3_lr1e-5	32.7378	98	0.00126744
x4	30.1533	95	0.00183



5.2. Analysis of Results

- **Impact of Upscaling Factor:** A clear inverse relationship exists between the upscaling factor and image quality. The **x2 model achieved the highest PSNR (36.48 dB)**, significantly outperforming the x3 (~33.1 dB) and x4 (30.15 dB) models. This is expected, as generating more pixels (e.g., 16 pixels from 1 for x4) is a more complex task.
- **Impact of Learning Rate:** For the x3 scale factor, a learning rate of **1e-5 was too low**, resulting in slower convergence and a lower final PSNR. The learning rates of **1e-3 and 1e-4 produced nearly identical, optimal results**, with 1e-3 achieving a marginally higher peak PSNR. A rate of 1e-4 is often considered a "safer" and more stable choice.

- **Correlation of Loss and PSNR:** The final loss values directly correlate with the PSNR scores. The best-performing model (x2) had the lowest loss, confirming that minimizing the MSE loss effectively maximized the PSNR.

5.3. Qualitative Results

The visual results below show a clear improvement from the blurry Bicubic LR input to the sharp SRCNN Output, which closely resembles the Original HR image.

6. Conclusion and Future Work

The SRCNN model successfully learned to reconstruct high-resolution images, significantly outperforming traditional bicubic interpolation. The experiments demonstrated that model performance is heavily dependent on the upscaling factor and requires careful tuning of the learning rate.

For future work, the following could be explored:

- Test the model on larger datasets like **DIV2K** to improve generalization.
- Evaluate the model's robustness by adding **noise** to the input images.
- Explore more complex architectures like **U-Net or ResNet** to potentially capture finer details.