Image Super Resolution using SRResNet

Presented by
Dibyanayan Bandyopadhyay (2111CS02)
Samrat Mukherjee (2111CS12)
Baban Gain (2111CS21)

Under supervision of
Dr. Arijit Mondal

Dept. of Computer Science and Engineering
Indian Institute of Technology Patna
Patna, India

10 April 2022

Content

Deep Learning and Image Super Resolution

Image Super Resolution

- Generate high resolution (HR) image from one or more Low Resolution (LR) images.
- HR images provide more details.
- Helps better results in other Computer Vision applications.
  - Medical image analysis, Satellite image analysis etc.

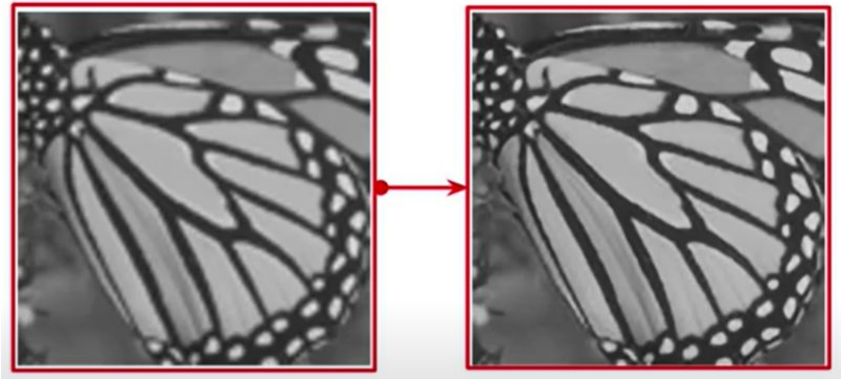
Why Deep Learning is a possible approach for this problem?

- Deep Neural Networks are universal function approximators.
- Can model the problem of Super Resolution as a function learning problem.

$$I_{LR} = D(I_{HR}, \sigma)$$

I_{LR}: Low Resolution Input image, I_{HR}: High Resolution input image, D: Degradation function, σ: noise

Example



(a)

(b)

Fig 1: Example of Image Super Resolution. High resolution image obtained by (a) SRCNN architecture. (b) Super Resolution via Repeated refinements.

SRResNet

- Super Resolution Residual Network.
- Introduced in the paper “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network” by C.Ledig et.al. in 2017.
- Inspired from the ResNet architecture “Deep Residual Learning for Image Recognition” by K.He et.al. 2015.
- Serves as a Generator for the Super Resolution Generative Adversarial Network (SRGAN).
- To train the SRGAN better, trained SRResNet is used to avoid local optima issue.

Architecture of SRResNet

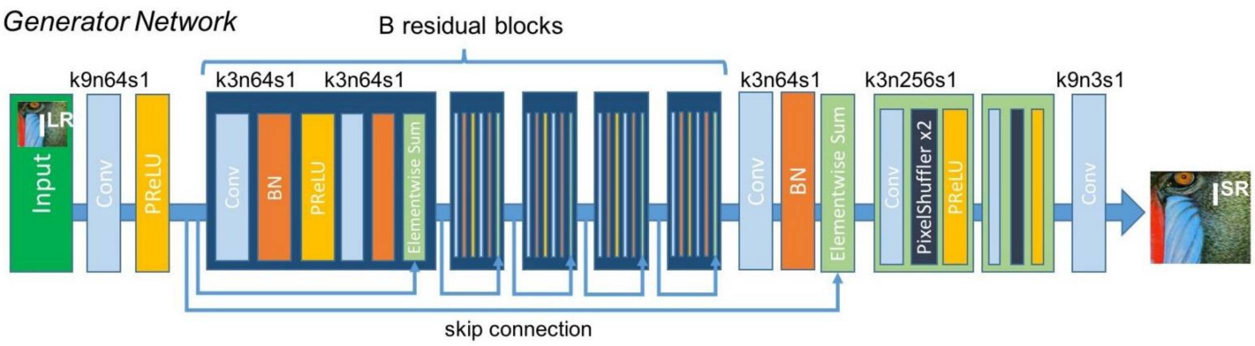


Fig 2: SRResNet model architecture

- Introduced Pixel Shuffle layers for upsampling the image.
- Used Parametric ReLU as activation function.
- 16 Residual blocks are used in the architecture.
- 3x3 filters are used in the Residual Blocks.
- Used MSE loss function for training.

Ideas behind proposed modifications

- **Use of Perceptual loss for training:** MSE chooses average of all possible images in the manifold.
    - Smoothens the output.
    - Not very sharp images.
- **Removing Batchnorm layers:** Inspired from the paper, “Enhanced Deep Residual Networks for Single Image Super Resolution” B.Lim et.al 2017

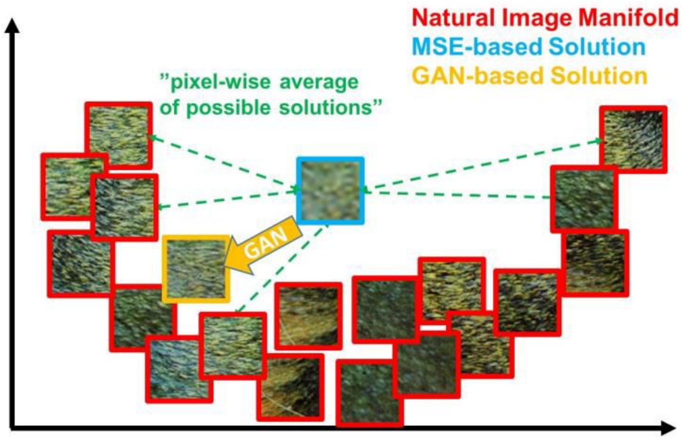


Fig 3: Issue with MSE as loss function and how Perceptual loss might help

Experiments

- We tried to code the entire architecture of SRResNet from scratch.
- Removed the BatchNorm layers from the Residual blocks.
- Trained the model using MSE loss and Perceptual Loss.
- Model specifications explored:
  - **Loss function:** Mean square error, **BatchNorm layer in Residual Blocks:** Yes
  - **Loss function:** Mean square error, **BatchNorm layer in Residual Blocks:** No
  - **Loss function:** Perceptual Loss, **BatchNorm layer in Residual Blocks:** Yes
  - **Loss function:** Perceptual Loss, **BatchNorm layer in Residual Blocks:** No

Results

- Limited scope of experimentations due to resource constraints.

Loss function	Batch normalization in the Residual Block	Epochs	PSNR (dB) (Set 5 dataset)	PSNR (dB) (Set 14 dataset)
Mean Square Error	Yes	50	30.608	24.702
	No	50	30.685	25.981

Loss function	Batch normalization in the Residual Block	Epochs	PSNR (dB) (Validation set)
Perceptual Loss	Yes	20	-0.2512
	No	20	9.213

Caveats

- **Space Constraints:**
    - ***Limitation***: Maximum 15 GB of GPU RAM offered by Kaggle and Google Colab.
    - ***Result***: Batch size limitation (max 2 or 3 examples per batch).
- More batch size and training time improves reconstruction.

References

- Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, C.Ledig, L.Theis, F.. Huszar, J. Caballero, A.Cunningham, A.Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W.Shi, CVPR 2016.
- Enhanced Deep Residual Networks for Single Image Super-Resolution, B.Lim, S. Son, H.Kim, S.Nah, K.M.Lee, CVPR 2017 workshop.
- Pytorch Perceptual Loss implementation: https://gist.github.com/alper111/8233cdb0414b4cb5853f2f730ab95a49
- Div2K dataset: Obtained from https://www.kaggle.com/datasets/sharansmenon/div2k
- SRResNet Pytorch implementation: https://www.kaggle.com/code/pastfi/srresnet/script
- Aladdin Persson SRGAN implementation: https://github.com/aladdinpersson/Machine-Learning-Collection