Text Image Super-Resolution

For improvement of OCR Accuracy

Department of Computer Science and Engineering
Indian Institute of Technology, Jodhpur

Presented by:
Qazi Sajid Azam (B16CS026)
Anurag Shah (B16CS034)

Mentor:
Dr. Gaurav Harit

Introduction

Abstract

We have designed a system that performs super-resolution of text images in order to improve the accuracy of OCR technologies.

Using deep learning methods, we are able to generate a high-resolution larger image from a smaller image, because the accuracy of OCR generally increases with larger, sharper images.

Concept of Super-Resolution

Super-Resolution refers to the task of recovering a high-resolution image from its low-resolution counterpart, while preserving the detail and visual fidelity expected from such a high-resolution image.

Modern state-of-the-art technologies implement deep learning to solve this problem, with varying levels of success.

Previous Work

Convolutional Neural Networks (SRCNN)

- Naive Initial approach
- Learn a mapping between the LR and HR images, from a large dataset of such pairs

Drawbacks:

- Loss functions not entirely indicative of the true visual quality
- Results in overly smooth textures
- Absence of high-frequency details
- Easily identified as fake

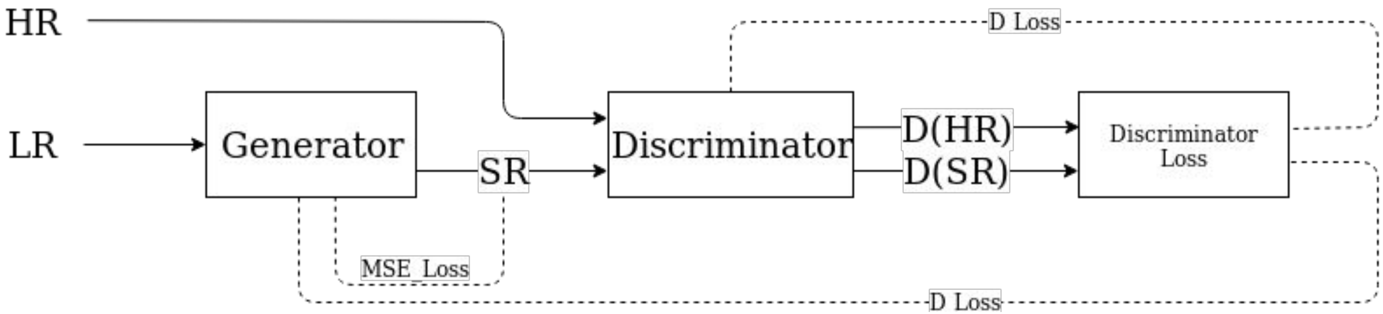
Chao Dong, Chen Change Loy, Kaiming He, Xiaoou Tang:
Image Super-Resolution Using Deep Convolutional Networks.

Generative Adversarial Networks (SRGAN)

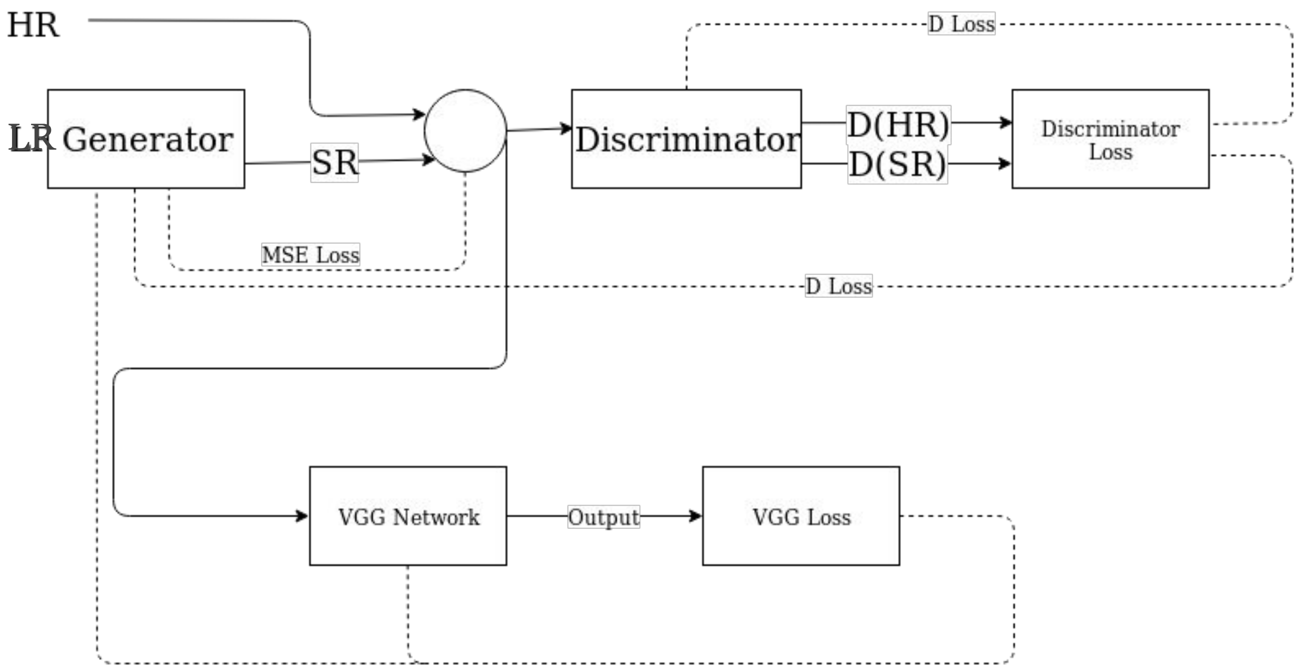
- Smarter approach
- Solves the problem of missing texture details
- Employs a discriminator network along with a generator network
- Loss function includes whether the image looks real or fake
- Results in highly photorealistic generated images

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi: **Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.**

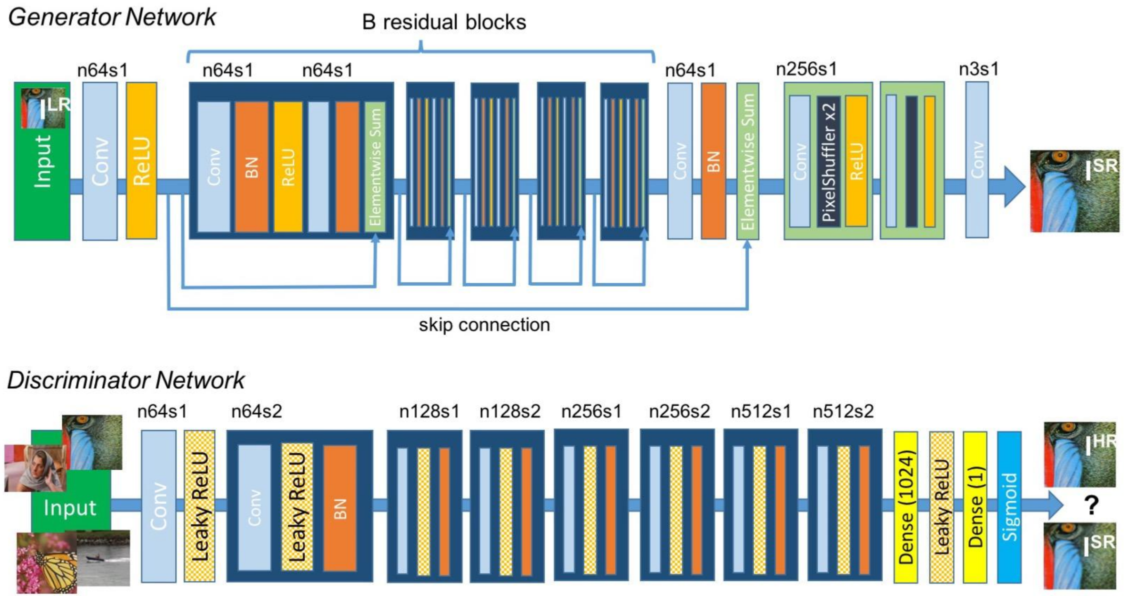
GAN Network Architecture



SRGAN Network Architecture



G and D Architectures



GAN as a Minimax problem

GAN can be seen as a minimax problem, which is represented in below equation.

$$min_{\theta_G} \; max_{\theta_D} \; E_{I^{HR} \sim p_{train}(I^{HR})} \left[log D_{\theta_D} \left(I^{HR} \right) \right] \\ + E_{I^{LR} \sim p_G(I^{LR})} \left[log \left(1 - D_{\theta_D} \left(G_{\theta_G} \left(I^{LR} \right) \right) \right) \right]$$

Work Done

Optimization for Text Images

- The original SRGAN is made for natural images.
- We have modified it to suit text images better by tweaking parameters
- Following improvements are made:
  - Tried different combinations of weights of loss components
  - Increase the weight of VGG loss, which relates to features
  - Include OCR evaluation of the results

Loss Function

We use a perceptual loss function with two components with empirical weights to get the overall loss function below.

$$l^{SR} = \underbrace{\underbrace{l_{\mathrm{X}}^{SR}}_{\text{content loss}} + \underbrace{10^{-3} l_{Gen}^{SR}}_{\text{adversarial loss}}}_{\text{perceptual loss}}$$

Loss Components: Content Loss

The content loss has further two components:

1. MSE loss (mean squared error)

$$l_{MSE}^{SR} = \frac{1}{r^2 WH} \sum_{x=1}^{rW} \sum_{y=1}^{rH} \left(I_{x,y}^{HR} - G_{\theta_G} \left(I^{LR} \right)_{x,y} \right)^2$$

2. VGG loss

$$l_{VGG/i,j}^{SR} = \frac{1}{W_{i,j} H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \left(\phi_{i,j} \left(I^{HR} \right)_{x,y} - \phi_{i,j} \left(G_{\theta_G} \left(I^{LR} \right) \right)_{x,y} \right)^2$$

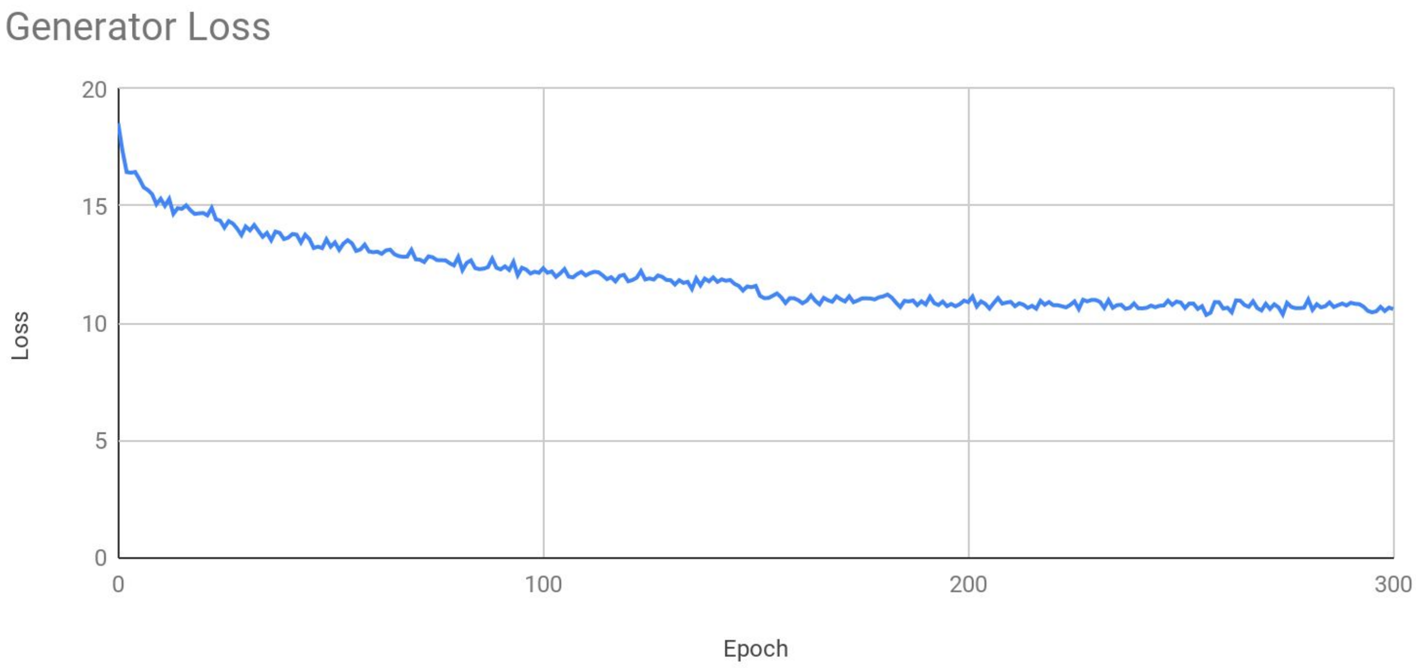
Loss Components: Adversarial Loss

This component of the loss encourages our network to favor solutions that reside on the manifold of natural images, by trying to fool the discriminator network.

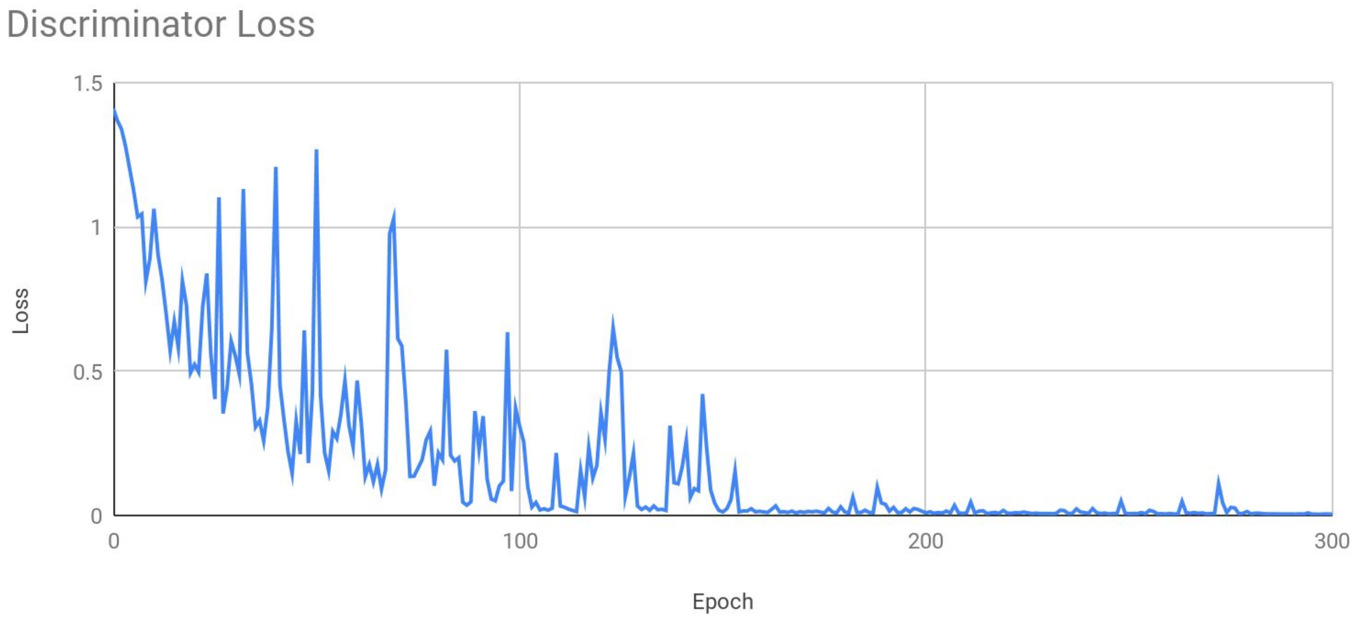
$$l_{Gen}^{SR} = \sum_{n=1}^{N} -log D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$$

Results

Loss over epochs: Generator



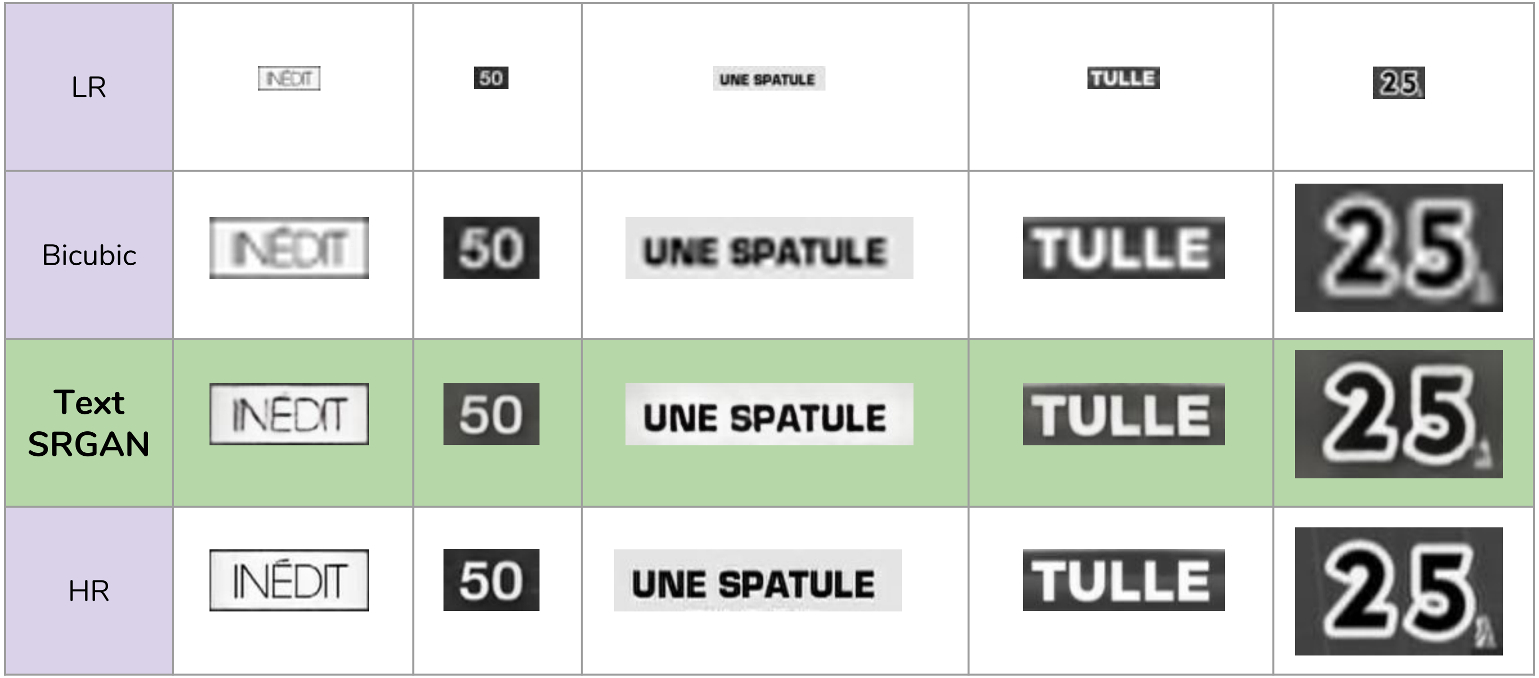
Loss over epochs: Discriminator



Comparison

	PSNR	MSE	SSIM
Bicubic	22.11959	618.1672	0.885283
SRGAN trained on DIV2K	21.45826	643.0672	0.864593
SRGAN trained on text	19.33736	833.2610	0.876426
Text SRGAN	**21.86047**	**475.5829**	**0.921915**

Some of the samples



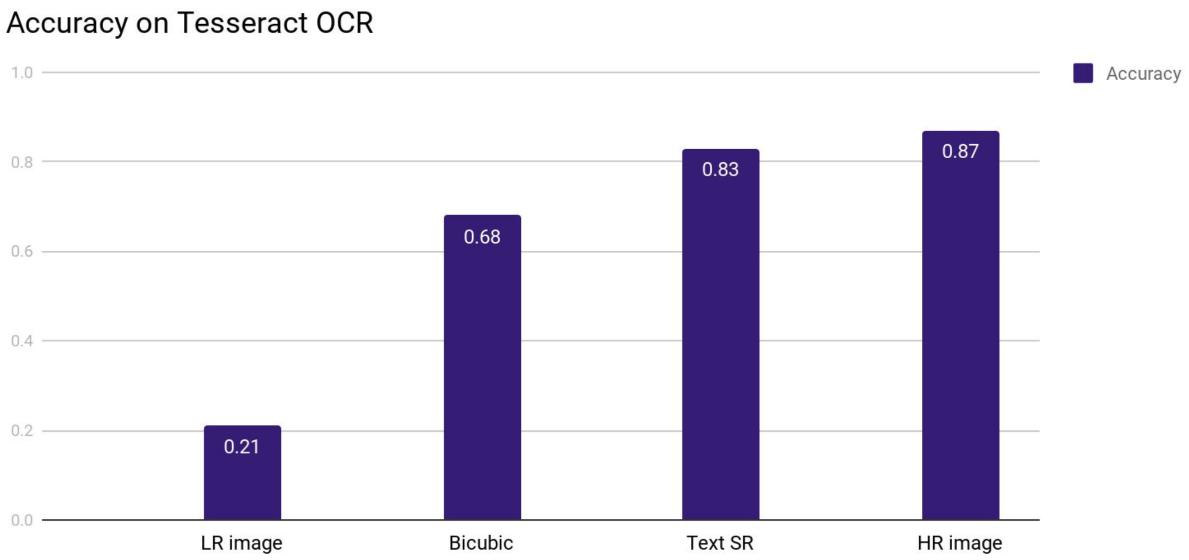
Output samples over the epochs

Animated GIF showing the changes every 25 epochs from epoch 0 to 175

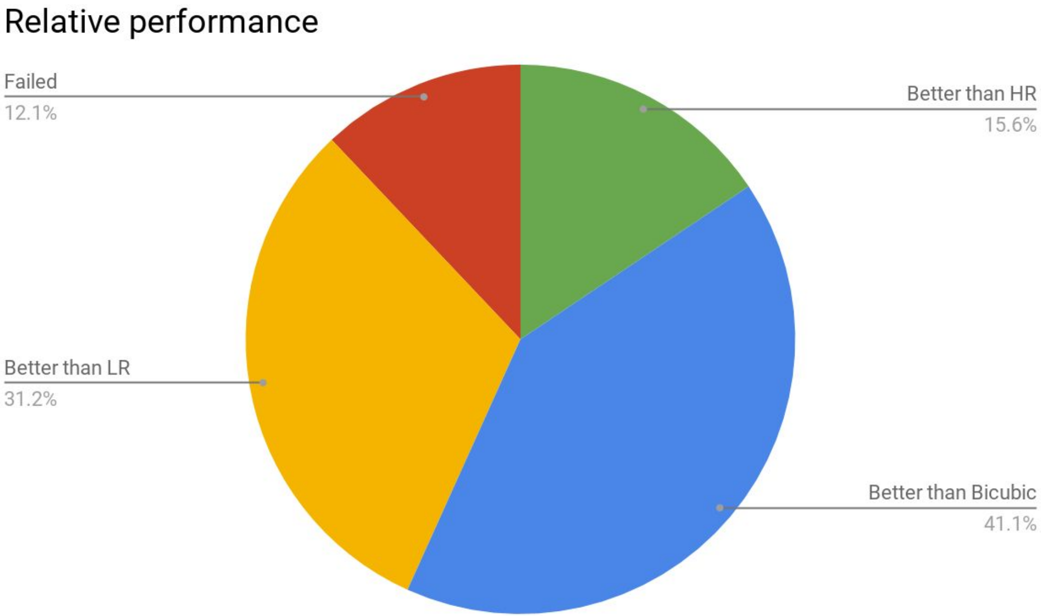
Final Result after 300 epochs

Ground Truth

Comparison of OCR Accuracies



Relative performance with OCR



Improvement in OCR Accuracy

Relative to HR	95.54%
Relative to Bicubic	121.3%
Relative to LR	390.0%

Future Work

Suggestions for Improvement of Results

- Further optimizations to loss function (replace VGG)
- Training on larger dataset for more number of epochs
- Including handwritten images dataset
- Improvement of network architecture

Thank You