Anime Super Resolution Using GANs

Jin Huang (huang86)

CS230 - Deep Learning, Stanford University

Introduction

- **Anime** images and videos are at most 1080p. The old classic ones are even at 480p.
- Anime resides in a lower dimension in terms of color and feature, usually with low frequency content. It's a **different and unique field** to focus on. Most importantly, I'm an anime fan.
- The best modern approach for super resolution problem is **SRGAN**. New solutions show up to improve the training of GAN, such as **WGAN with gradient penalty**.

Data

- Only images larger than 512 x 512.
- 11328 random images from **kaggle dataset** https://www.kaggle.com/mylesoneill/tagged-anime-illustrations/home.
- Train/Dev/Test set sizes are 10816/256/256.

Fig 1. Dataset sample

Preprocessing

- CenterCrop to 384 x 384
- RandomCrop to 128 x 128 (SRGAN) or 96 x 96 (SRWGAN-GP) - Original HR
- Bicubic downsample to 32 x 32 (SRGAN) or 24 x 24 (SRWGAN-GP) - LR Input

Method

SRGAN Discriminator

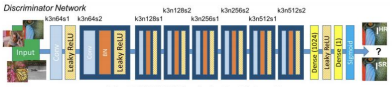

Fig 4. SRGAN Discriminator Network

Fig 6. SRGAN Discriminator Network Details

SRWGAN-GP Discriminator

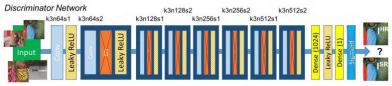

Fig 5. SRWGAN-GP Discriminator Network

- Input & output pixel values [0, 1].
- SRWGAN-GP has no BatchNorm layers.
- SRWGAN-GP has no sigmoid.

* Batch normalization creates correlation between samples in the same batch. It impacts the effectiveness of the gradient penalty.

SRGAN Discriminator Loss

$$-\frac{1}{m_{real}}\sum_{i=1}^{m_{real}} y^{(i)}_{real}.\log(D(x^{(i)})) - \frac{1}{m_{gen}}\sum_{i=1}^{m_{gen}}(1-y^{(i)}_{gen}).\log(1-D(G(z^{(i)})))$$

SRWGAN-GP Discriminator Loss

$$\underbrace{\mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})] - \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)]}_{\text{Original critic loss}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1)^2]}_{\text{Our gradient penalty}}$$

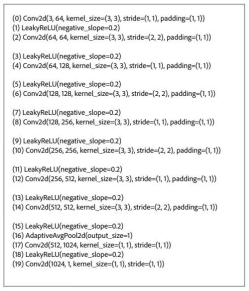

Fig 7. SRWGAN-GP Discriminator Network Details

Method

Generator

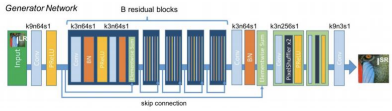

Fig 2. Generator Network

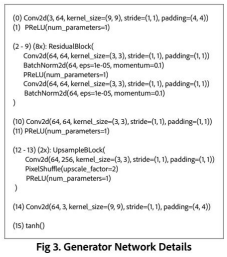

Fig 3. Generator Network Details

- Input & output pixel values [0, 1].
- Tanh as the last layer.
- Using 8 residual blocks, not 16.

Generator Loss

Loss = MSE_loss + λ * adversarial_loss
* pre-trained model **NOT** used

Generator Adversarial Loss

SRGAN

$$-\frac{1}{m_{gen}}\sum_{i=1}^{m_{gen}}\log(D(G(z^{(i)})))$$

SRWGAN-GP

$$-\frac{1}{m_{gen}}\sum_{i=1}^{m_{gen}}D(G(z^{(i)}))$$

Training

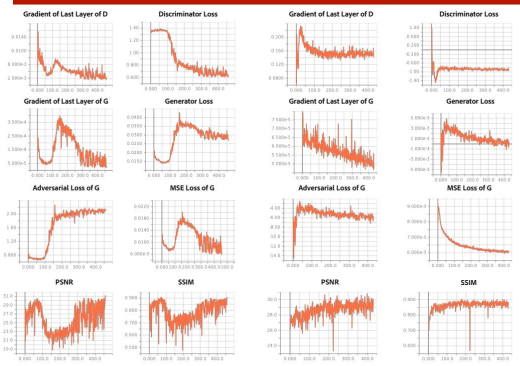

Fig 8. SRGAN Training

Fig 9. SRWGAN-GP Training

HyperParameters

- 500 epochs.
- Default initializations from PyTorch.
- ADAM, betas=(0.9, 0.999), eps=1e-8.
- SRGAN uses learning rate 1e-3, SRWGAN-GP uses 1e-4.
- SRGAN uses λ = 1e-2 in Generator loss, SRWGAN-GP uses 1e-3.
- SRGAN has label smoothing, Real (0.85, 1.10), Fake (0, 0.15), and noisy labels.
- SRGAN has input 32 x 32 and batch size 64, SRWGAN-GP has 24 x 24 and 32.

Findings

- SRWGAN-GP's MSE loss decreases smoothly.
- SRWGAN-GP has stable gradients.
- SRWGAN-GP converges faster.
- SRGAN gets higher PSNR and SSIM at peak.

Results

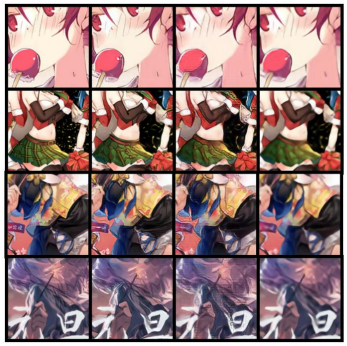

Fig 10. Original HR / SRWGAN-GP / SRGAN / Deep ResNet

* Input LR images are 4x4 downsampled from Original HR ones. SR mages are similar with original HR images when the model is working.

- Deep ResNet gets very blurry results with high PSNR and SSIM.
- SRGAN results are more blurry than SRWGAN-GP ones by average.
- SRGAN fails in more cases. SRWGAN-GP produces good results even when SRGAN fails. See the last row in Fig 10.

Conclusions

- Anime and realistic photos are on different distributions. Anime Super Resolution is a different problem from realistic photo Super Resolution.
- SRGAN is working on Anime Super Resolution without features from per-trained models.
- SRWGAN-GP overall outperforms SRGAN on Anime Super Resolution.

Future Work

- Explore more GAN training tricks.
- Experiment with other GANs.
- Work on Anime video Super Resolution. Use RNN to get consistent video frames.