Text Image Super-Resolution Reconstruction Using GAN

Haoxin Liu
ShanghaiTech University
liuhx@shanghaitech.edu.cn

Zhongyi Cai
ShanghaiTech University
caozhy@shanghaitech.edu.cn

Wenfu Xia
ShanghaiTech University
xiawf@shanghaitech.edu.cn

Lixuan Chen
ShanghaiTech University
chenlx1@shanghaitech.edu.cn

Zhaozheng shen
ShanghaiTech University
shenzhzh@shanghaitech.edu.cn

Abstract

Image super-resolution reconstruction technology has made rapid progress in recent years from using CNN to GAN. As one of its subfields, text image super-resolution also make some breakthrough by introducing neural networks into the reconstruction process and provide great help for OCR recognition. In our work shown in this paper, we apply generative adversarial network to reconstruct low-resolution image into high-resolution image, during which one new weighed MSE loss function is used in training process. Beside this, we propose a way of padding which will be used for text images and design a new evaluation mechanism using OCR tool for the whole model finally.

1. Introduction

Image Reconstruction is a classic application of computer vision. Its purpose is to recover lost information by using the existing information of the image and it can be used for recovering lost information in old photos, image de-watermark and video text removal. In short, image reconstruction is the process of filling information into the information defect area on the image and making the observer can not perceive that the image has been damaged or has been repaired. In the field of image processing, the information that low-resolution images can provide is very limited. People always expect to obtain high-resolution images because of their high detail, rich texture, and clear images and a large amount of useful information can be captured from it. One significant application is to repair old text images, such as unreadable newspaper, making blurred images clear. This project will focus on text images' super-resolution reconstruction and try to work restoring the blur text images provided by ICDAR2015 competition.

Generative adversarial network(GAN) is widely adapted in image super resolution. The generator network learns to generate plausible data. The generated instances become negative training examples for the discriminator, and the discriminator learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results. We are using SRGAN as our neural network, which is trained and tested on ICDAR2015 Text Image dataset.

We also propose an image padding method which will be applied after the network is trained. Before inputting an test image into the network, we recursively enlarge the image by padding the boundaries with a strip width. In addition, we changed the loss function to capture more texture information.

Here is an example show the process of our work in this paper:

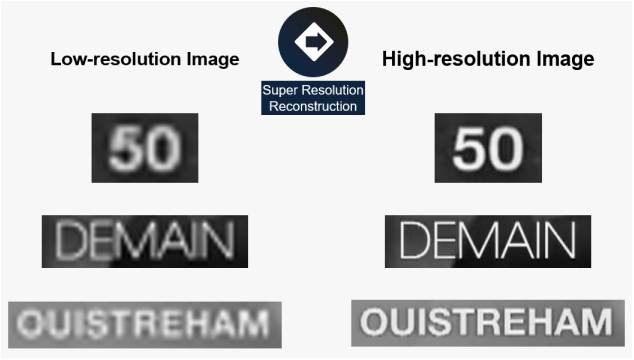

Figure 1. Process Example

2. Related Work

Our work mainly relates to two papers: Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network by Ledig et al. [1] Another paper is CNN-Based Text Image Super-Resolution Tailored for OCR by Haochen et al. [3]

2.1. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network

In this paper [1], the authors present SRGAN, a generative adversarial network (GAN) for image super-resolution (SR). They propose a perceptual loss function which consists of an adversarial loss and a content loss and an extensive mean-opinion-score (MOS) test shows hugely significant gains in perceptual quality using SRGAN.

Because SRGAN shows great performance in image super-resolution and GAN can generate highly detailed textures, we wonder if it will also perform well in the text image and we are inspired to apply SRGAN into our work.

2.2. CNN-Based Text Image Super-Resolution Tailored for OCR

In this paper [3], authors concern more on the OCR performance of the reconstructed images. They propose a new loss function when training CNN for text image SR to facilitate OCR. Also, a simple yet effective image padding method is introduced into refining the image boundaries during SR. Experimental results show that the OCR accuracy achieved by their optimized model is close to the OCR accuracy achieved by directly using the original high-resolution images, which shows the success of their work.

Although this model has already preformed well, the CNN network is very simple and we wonder if we could use a more efficient network which can generate highly detailed textures and could get better preformance.

3. Our Approach

3.1. Image Padding

Since the size of text images in the dataset are much smaller than the normal images, the problem, missing of boundary information, will become more serious. We propose an image padding method which will be applied after the network trained. Before inputting an test image into the network, we recursively enlarge the image by padding the boundaries with a strip width. Also, the network output is cropped at the center to produce the final super-resolved image that has the same size as the original.

Our padding method is designed that the pixel value of each padded pixel is decided as the average of pixel values of the nearest to the padded one and are on the image boundaries. We define that the nearest pixels are those within a strip width and we deal with corner pixels carefully. For example, for the padded pixel (w, h), if the padding width is 2, then the nearest pixels are those within $(w-2, h-2)$ to $(w+2, h)$. The Fig. 2 shows an intuitive explanation.

3.2. Loss Function

In the SRGAN, the loss function divides into two parts: adversarial loss and content loss.

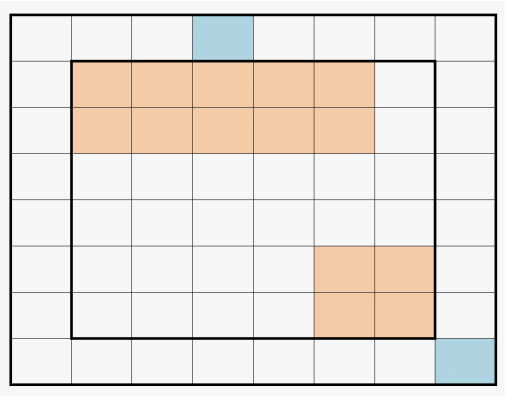

Figure 2. Example of padding. Blue pixels are the padded pixel and orange pixels are the nearest pixels.

$$\underbrace{l^{SR}}_{perceptualloss} = \underbrace{l^{SR}_{X}}_{\text{content loss}} + \underbrace{10^{-3} l^{SR}_{Gen}}_{\text{adversarial loss}} \quad (1)$$

In our Approach, we add a WMSE(Weighed MSE) loss and a PSNR loss.

3.2.1 Content loss

The content loss also divides into two parts: MSE loss and VGG loss. However, while achieving particularly high PSNR, solutions of MSE optimization problems often lack high-frequency content and the textures information. Therefore, SRGAN adds a VGG loss based on the ReLU activation layers of the pre-trained 19 layer VGG network. VGG lose is the euclidean distance between the feature representations of a reconstructed image and the groundtruth image. In our approach, we keep this content loss and we reduce its weight.

3.2.2 Adversarial loss

SRGAN adds the generative component of our GAN to the perceptual loss. For our adversarial loss, we keep it same as SRGAN.

3.2.3 Weighed MSE loss

The pixel-wise MSE loss is calculated as:

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

Most of the previous works adopted MSE as the loss function. Actually, MSE minimization is equivalent to PSNR optimization. However, PSNR is not a good indicator of the OCR accuracy. MSE treats every pixel equally

important but text regions usually feature high contrast edges which represent high-frequency component of an image. Moreover, OCR accuracy depends highly on the high-frequency image details in the text regions of text images. Therefore, in our approach, we add a new loss function: Weighed MSE(WMSE) loss which thinks some pixels are more important.

$$WMSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} ||I(i,j) - \hat{I}(i,j)||^2 \times f[grad(i,j)]}{mn} \tag{3}$$

We use a Sobel operator to calculate the gradient magnitude map of the original image. Fig .3 is the x-direction gradient and y-direction gradient. Then we use a map $f[\cdot]$ which is a monotonously increasing function to convert gradient magnitude into weight.

Figure 3. The up image is x-direction gradient of the original image and the blow image is y-direction gradient of the original image.

We choose this function among $f[x] = x^2$, $f[x] = x$ and $f[x] = \sqrt{x}$ and when $f[x] = x$, the performance is best which is described detailedly in the result section.

When we implement the WMSE, we set $g[x] = \sqrt{f[x]}$ to make our calculation more easier and the WMSE will transform to

$$WMSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} \|I(i,j)\times f[grad(i,j)] - \hat{I}(i,j)\times f[grad(i,j)]\|^2}{mn} \tag{4}$$

3.2.4 PSNR loss

We also add the PSNR loss and the PSNR loss is

$$PSNR = 10 \cdot \log_{10}(\frac{MAX_I^2}{MSE}) \tag{5}$$

3.2.5 SSIM loss

We also try to add the SSIM loss. However, SSIM loss is almost stable and we think it is useless thus we remove it.($c_1 = 2.55^2$,$c_2 = 7.65^2$)

$$SSIM = \frac{((2 * \mu_x * \mu_y + c_1) * (2 * \sigma_{xy} + c_2))}{((\mu_x^2 + \mu_y^2 + c_1) * (\sigma_x^2 + \sigma_y^2 + c_2))} \tag{6}$$

3.3. OCR

How to design an effective and fair evaluation mechanism is crucial for evaluating a project. The traditional evaluation mechanism of super-resolution image is usually gathering a group of voters to vote for the reconstructed image, which requires a lot of manpower and resources and is subjective to some extent. Under this circumstance, our project uses OCR to evaluate the generated images. It first imports a OCR package (pytesseract) in python, uses it to identify the characters in the the high-resolution image and takes them as the groundtruth. Then the super-resolution reconstructed images and low-resolution images were identified and compared with the ground truth to calculate the accuracy. Finally, it adds up the accuracy of each super-resolution reconstructed images and calculates the average accuracy.

4. Experiment

4.1. Dataset

Our text image super resolution model is trained and tested on the set of ICDAR(ICDAR2015-TextSR-dataset), which aims to motivate research around Super-Resolution (SR) and its application in the specific context of Text Images.

4.2. WMSE

Firstly, we choose the form of weighting function and the results are showed in Table 1. We observed that the map function which is $f(x) = x$ perform best.

Function	Average GEN Accuracy(%)
sqrt	66.51
mul	6.87
self	**74.88**

Table 1. Map function choosed in WMSE, the result shows that self is the best monotone increasing function.

Then we apply the map function as $f(x) = x$ and choose weight of WMSE. The results are showed in Table 2. We observed that the OCR Accuracy is as high as 80.89%, while the accuracy decrease with the increase of weight because of the noise.

WMSE para	Avg.GEN Accuracy(%)	Avg.LRI Accuracy(%)
0.1	**80.89**	64.67
1	74.05	64.67
3	71.30	64.67

Table 2. Result on different weight for WMSE.

4.3. Image Padding

To capture boundary information, we propose an image padding method which will be applied after the network trained. Before inputting an test image into the network, we recursively enlarge the image by padding the boundaries with a strip width. We study whether changing the padding boundary operations will make much difference. Table 3 shows the OCR accuracy of the four different methods. We can see that Mean padding has the best performance so we choose it as our padding algorithm.

Method	Average GEN Accuracy(%)
No padding	75.17
Zero padding	79.29
Reflect padding	74.01
Mean padding(Ours)	**80.89**

Table 3. Image padding with different padding boundary operations. Ours is better.

4.4. OCR Result

Figure 4 shows that the OCR accuracy of the 15th epoch is the highest one, so we choose this model as our final result.

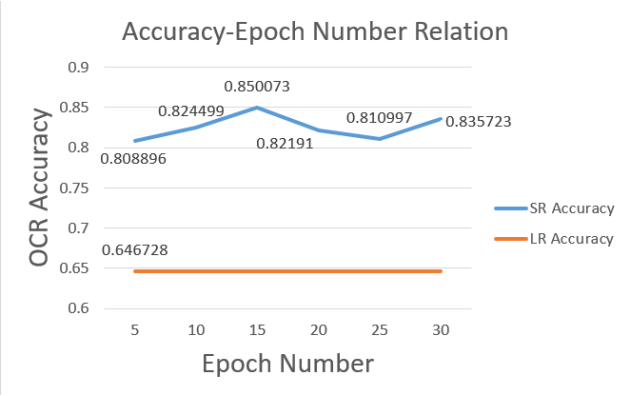

Figure 4. OCR results for different epochs

5. Future Works

Perceptual loss network has a component of the VGG loss, which can extract good features for natural images. While our project is focused on the text images, which strongly depend on the extraction of texture features. So the feature extraction network may not be limited to VGG. Meanwhile, we might using a network trained on test images, which may lead to a better result.

Recognize text of the image and text prior information are only used in the OCR evaluation mechanism. Since text images contains amount of semantic information, they produce useful information on generating super resolution images. In this way a model based on probability or LSTM may be helpful for optimizing the generator to produce content-aware text images, similar to the recognition network used in [2] .

In addition, image sharpening and image denoising may be helpful for enhancing edge information. We may try to sharpen the super resolution output to produce better visual effects later.

References

[1] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2016.

[2] Wenjia Wang, Enze Xie, Peize Sun, Wenhai Wang, Lixun Tian, Chunhua Shen, and Ping Luo. TextSR: Content-aware text super-resolution guided by recognition, 2019.

[3] Haochen Zhang, Dong Liu, and Zhiwei Xiong. Cnn-based text image super-resolution tailored for ocr. pages 1–4, 12 2017.

[4] Wang, Xintao, et al. Esrgan: Enhanced super-resolution generative adversarial networks. Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[5] Chu, Mengyu, et al. Temporally Coherent GANs for Video Super-Resolution (TecoGAN). arXiv preprint arXiv:1811.09393 (2018).

[6] Wang, Xintao, et al. Recovering realistic texture in image super-resolution by deep spatial feature transform. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[7] Sajjadi, Mehdi SM, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. Proceedings of the IEEE International Conference on Computer Vision. 2017.