Super Resolution for Sports Images
Computer Vision

James Liljenwall

June 9, 2019

Github Link

Introduction

With increasingly higher definition cameras becoming readily available, and screens being able to contain more pixels, the sizes of our images and videos have continued to increase. As these images and frames become larger, it becomes increasingly important to compress them in order for them to be easily transmitted and stored. However, there are significant limitations to the ability to perfectly compress an image, so if want to compress the images more, we need to allow for the new image to differ from the original. Super Resolution is one of the solutions to reconstructing the higher resolution images from a lower quality one. In this method, we use the fact that many of the pixels in an image can be closely estimated from those surrounding it, and using this we can remove many of the pixels.

Related Work

Trying to perform super resolution using Deep Learning is a fairly new concept, an has only been around since the early 2010's. Other forms of Super Resolution has been around for a while, such as bicubic upsampling. In the beginning, most of the research has been focused on using Deep Convolutional Neural Networks in order to try and find the implied pixels. However, currently, most of the research for still-images revolves around using a Generative Adversarial Network to create these images.

A Generative Adversarial Network uses a combination of two neural networks that are trained at the same time. The first one is a generator network, which attempts to create a high resolution image based on one that was passed in. The second network is a discriminator, which attempts to distinguish whether or not the images created by the generator are true images or fake. The loss functions of the Generator is thus dependent not only on how close it is to the high quality original image but also whether or not it was able to fool the discriminator.

This Generative Adversarial model for Super Resolution was first presented by Ledig Et Al. in their paper *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network* as SRGAN, and has since become the standard model for Super Resolution. This model has been re-implemented numerous times, and has been improved with the use of Residual-in-Residual Dense Blocks, which uses SRGAN's model without Batch Normalization, which was proposed by Xintao Et Al. in *ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks*

Using an implementation of SRGAN, I will be training the model using the images from sporting events in order to see if I can improve the quality of the generated images by training on a smaller subset of types of images.

Dataset

Super Resolution, is used to infer details of an image, which generally is caused by one of two factors: the device does not have high enough resolution to catch all of the detail (as with a cell phone camera), or the observed images lack the detail due to physical limitations (such as light scattering through the atmosphere). For this project, we will be focusing exclusively on the case where the camera lacks the resolution to capture

all of the images.
The dataset that is used comes from Li-Jia Li and Li Fei-Fei, and consists of over a thousand images of sporting events, such as Rowing, Polo, Snow Boarding, Sailing, Badminton, Bocce, Croquet, and Rock Climbing. For each of the images, I cropped it to be 512x512 pixels. I used these images as the high resolution labels, and created the lower resolution images by scaling these images down to 128x128 pixels. I split the data into 80/10/10 train/dev/test sets. In order to augment this dataset, when training, randomly cropped and flipped images were also used.

Methods

baseline

For my baseline, I will be comparing the results to what was achieved with the pre-trained SRGAN, this model has been trained on many more examples, and for longer than what my model was trained on, so I expect it to be superior. I am planning on analyzing how close it gets to the value

SRGAN

I trained the SRGAN located here *https://github.com/brade31919/SRGAN-tensorflow*. For the SRGAN model, there were two Networks that were trained: a Generator and a Discriminator.
The Generator network was a convolutional neural network composed primarily of 8 residual blocks. Each Residual Block was composed of a 3x3 filter convolution layer, batch norm, Relu, 3x3 filter Convolution, batch norm, and then an element-wise Sum.
The Discriminator Network is a convolutional neural network, which primarily consisted of 8 blocks. Each block contained a 3x3 filter convolutional layer, a batch norm layer, and then a leaky relu layer.

Training

While Training these networks, the Generator was first trained, as we used a pre-trained discriminator. Then We trained both networks in conjunction. For these networks, we used Mean Squared Error. In order to train the GAN, we also included Adversarial loss, which was the Ratio times the log of the output of the discriminator.

hyper parameters

The First hyperparameter I tuned was the learning rate. I tried training with a learning rate of .001, .0001, and .00001. After testing, I found that with a learning rate of .001 the training loss, never fell below .15, a learning rate of .0001 resulted in a training loss of .03, and the training loss with a learning rate of .00001

was .07. As a result for all future Training, .001 was used for the learning rate.
The Next Hyper-parameter that I tuned was the Batch Size of the dataset. I tested batch sizes of 4, 8, 16, and 32. For each batch size, I trained the algorithm for an equal amount of time, and then used the dev set to determine which one performed the best.

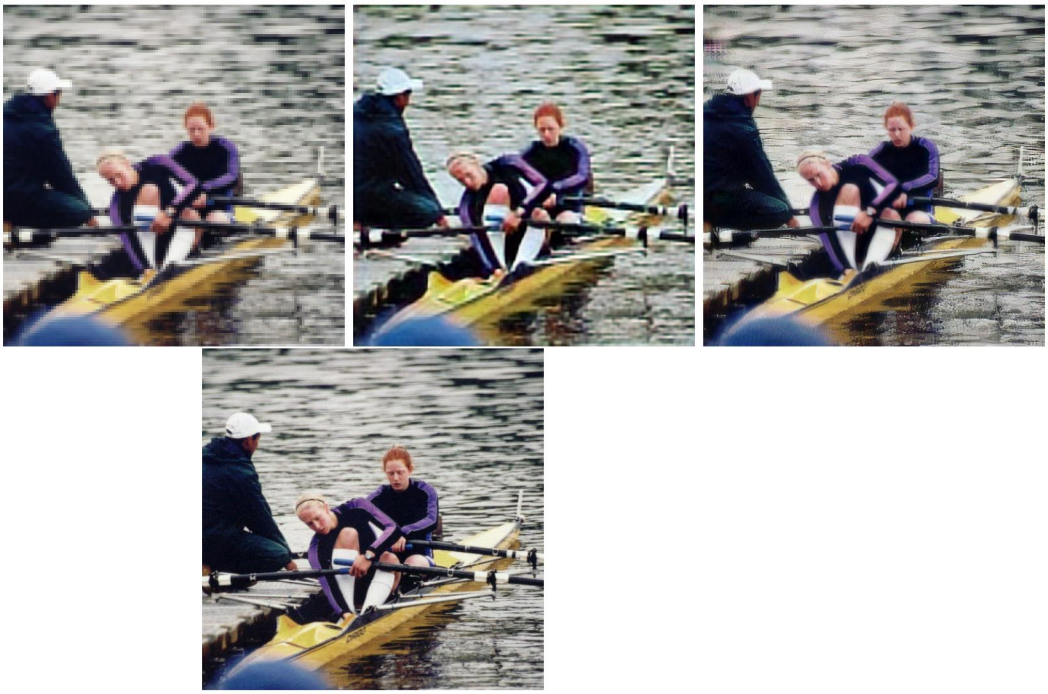
Batch Size	4	8	16	32
PSNR	50.04	50.308	50.687	50.25

From this we can see that the batch size of 8 resulted in the highest PSNR, which is what we went for in our final testing.

Results

After Testing the SRGAN model trained on the smaller subset of sporting images, we came up with the following results: Here is an example of the output from the Netowrk

model	SRGAN (sport trained)	SRGAN (pre-trained)
PSNR	50.621	52.187



The Images from left to right: input image, output from sports SRGAN, output from pre-trained, true image

Looking at the Images we can see that our SRGAN did significantly better than the original input, and was fairly close to the pre-computed image.

Conclusion/Future Work

Overall, the SRGAN trained with only on images from Sporting events did worse than the one that was pre-trained. However, the images was still fairly close, and it shows that the Network can achieve high performance on a smaller population, with a specific task. With images, it appears that training on more examples is always going to be better.
For future work, it would be useful to get a larger database of sports images, and try to see at what point

the network achieves the same performance, and at what point it performs better. Additionally, it could be useful to see if it could achieve better performance trying a different degree of upscaling (instead of just 4x as was done in this project). Finally, It would be useful to see if training this for a longer period of time, would also have enabled it to perform better. The pre-trained model was trained for weeks, which was impractical in this situation, so it would be interesting to see if training it for longer would improve the performance significantly.

sources

https://github.com/brade31919/SRGAN-tensorflow
http://openaccess.thecvf.com/content_cvpr_2018/papers/Jo_Deep_Video_Super-Resolution_CVPR_2018_paper.pdf
C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, accepted at CVPR (oral), 2017.
Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A. Image-to-Image Translation with Conditional Adversarial Networks CVPR, 2017.
pix2pix-tensorflow D.-T. Dang-Nguyen, C. Pasquini, V. Conotter, G. Boato, RAISE – A Raw Images Dataset for Digital Image Forensics, ACM Multimedia Systems, Portland, Oregon, March 18-20, 2015".
Li-Jia Li and Li Fei-Fei. What, where and who? Classifying event by scene and object recognition . IEEE Intern. Conf. in Computer Vision (ICCV). 2007 (PDF)