Astronomical Image Colorization and Super Resolution using Residual Encoder Networks and GANs

Gao Xian Peh
Department of Computer Science
Stanford University
Stanford, CA 94305
gxpeh@stanford.edu

Kai Marshland
Department of Computer Science
Stanford University
Stanford, CA 94305
kaimarsh@stanford.edu

Abstract

Ordinarily processing raw images from the Hubble Space Telescope is a time-consuming manual process. We investigate how to do two core tasks in the image processing pipeline automatically, namely colourization followed by super-resolution. We connect a residual encoder network for colourization to a generative adversarial network for super-resolution so as to produce high-quality, coloured final images.

1 Introduction

A huge number of raw images lie unprocessed and unseen in the Hubble Legacy Archives. These raw images are typically low-resolution, black and white, and unfit to be shared with the world. It take tens, or even hundreds of hours to process them (1). This requires far more resources than NASA has available, so they ask for help from amateurs (2).

This processing is necessary because astronomers often struggle to distinguish objects from the raw images. Random and systematic noise from the sensors in the telescope, changing optical characteristics in the system, and noise from other bodies in the universe all make processing necessary. Furthermore, colourization is needed to help highlight small features that ordinarily wouldn't be able to be picked out against the noise of the image.

This processing of images is so time consuming that much of the Hubble data is only rarely seen by human eyes. This problem is only likely to get worse. Not only is new data being continuously produced by the Hubble Space Telescope, but also new telescopes are soon to come online. When Hubble's successor, the James Webb Space Telescope, at last launches, it will produce vastly more data than Hubble ever did. We seek to simplify this process, using artificial colourization and super-resolution. This will make it dramatically easier for astronomers to visually identify and analyze objects in the Hubble dataset.

2 Related work

2.1 Image Colorization

Our project is largely inspired by Ryan Dahl's CNN-based system (3) for automatically colorizing images. Dahl's system applies transfer learning from the ImageNet-trained VGG-16 network (4) and is integrated with an autoencoder system with residual connections that merge intermediate outputs produced by the encoding portion of the network which includes the VGG16 layers and the decoding

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

part of the network. These residual connections link downstream network edges with upstream network edges, enabling gradients to propagate faster through the network.

Residual Encoder Networks was first introduced by Kaiming He et al. (5) in 2015 as a method to easily train and optimize deeper neural networks to obtain better model accuracy and performance. This was previously not possible as deeper networks often have poorer performance beyond a particular depth limit because it was difficult for the network to learn simple functions of its inputs. Residual connections reduce the time taken for convergence and enables the training of deeper neural networks. We use Residual Encoder Networks for colorization as it is one of the state-of-the art methods in image colorization.

2.2 Image Super-Resolution

Generative Adversarial Networks (GANs) was first introduced by Ian Goodfellow (6) in 2014. By having two neural networks (Discriminator and Generator) compete with each other in a zero-sum game, GANs enable the production of synthetic images that look superficially authentic to the human eye. Christian Ledig et al. (7) introduced the SRGAN (Super Resolution GAN) in 2017, which used a SRResNet (Super Resolution Residual Network) as its generator to produce realistic-looking images with an upscaling factor of 4x. The SRResNet improves upon the CNN by using a Residual Network Architecture (8) that is able to manage deeper networks with the use of skip connections while avoiding the vanishing gradient problem often present in vanilla neural networks. We use the SRGAN as it is the state-of-the-art method for image super-resolution as compared to bicubic interpolation or traditional sparse-encoding based methods.

The standard evaluation metrics for image super-resolution are Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR) (9; 10). However, some papers argue that these metrics only evaluate pixel-wise differences but fail to capture the visual similarities between the SRGAN output and the original high-resolution image. Hence, even though we report the MSE and PSNR as part of our quantitative results, we train our SRGAN using perceptual loss (11; 12) on a pre-trained VGG19 (13) network on ImageNet instead.

3 Dataset and Features

We initially started by using data scraped from the Hubble Legacy Archive. Using puppeteer (headless Chrome), we scraped some of the tens of thousands of colourized images it has available. Since the sky is large, and the Hubble Legacy Archive website slow, we filtered to only images of the Messier 101 (M101) galaxy. As a large, nearby, and well-studied galaxy, the tens of millions of raw M101 images are representative of other Hubble images, and has been the subject of some of the largest colourization projects (14) making it a good source of data. However, we soon found that the Hubble Legacy Archive images were still too grainy to be useful. For training our networks, particularly the image super-resolution Generative Adversarial Network (SRGAN), we needed high-resolution, well-coloured images.

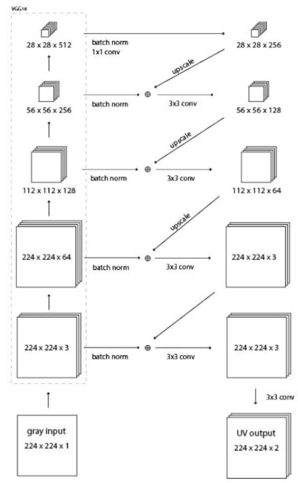
To that end, we scraped the Hubble Heritage project instead. The Hubble Heritage project releases by far the highest-quality astronomical images available. These are all laboriously stitched together, colourized, and processed to eliminate noise. Hubble Heritage then selects the best, most striking of these for public release. However, there were only 126 of these images. To increase the amount of data we had, we scraped images from the main Hubble website as well. This provided an extra approximately 2000 images.

We found that the sizes of the images varied dramatically, with some being as much as 40,000 pixels across. To augment our dataset, as well as to ensure uniform input, we cropped out tiles of 2040x1356 pixels. When an image was less than this size, we threw it out. If this size did not fit evenly into the image, we took overlapping crops, as a rudimentary form of data augmentation. We additionally converted all our images to PNG in order to be compatible with our pipeline. After this process was complete, we were left with 10,027 images. We took 80% of the dataset for our training set, 10% of the dataset for our dev set and the remaining 10% of our dataset for our test set.

4 Methods

4.1 Image Colorization

We adapted a Residual Encoder Network for image colourization, as described above, from the Guo implementation (15). The architecture was as follows (image credit (3)).



We calculated loss as the average of three different distances. The first was the squared difference between the generated image and the original. The second was the squared between a blurred version of the generated image and a blurred version of the original, and the third was calculated an even more blurred version. The first blurring was done with a 2D convolution using a 3x3 Gaussian blur kernel, and the second blurring was done with a 5x5 Gaussian blur kernel. The rationale is that we want colour to be preserved across blurring; without this blurring, colour would change much more rapidly, giving more patchwork results.

A key point about the network is that it made use of a pre-trained VGG-16 network. By taking advantage of this pre-trained network, it can much more quickly converge to a high quality result. The higher levels of the VGG-16 network were then fed through the residual encoder.

4.2 Image Super-Resolution

Our project is largely inspired by Christian Ledig's SRGAN paper (7) and Dong et al. (16) implementation of SRGANs using TensorLayer. Transfer learning was also used by using pre-trained weights from the VGG-19 network (13). As seen in Figure 1, we observe that the SRResNet acts as the generator of the SRGAN model and it contains both residual blocks and skip connections. Within each residual block, there are two convolutional layers followed by a Batch Normalization layer and a parametric ReLU layer. Finally, the image is then upscaled by 4 times using two sub-pixel convolutional layers. (17)

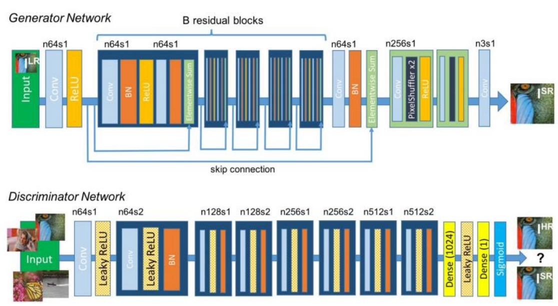


Figure 1: SRGAN model: SRResNet Generator Network and Discriminator

The generator's goal is to produce high resolution images to fool the discriminator of the GAN into thinking that it is receiving real instead of generated images. On the other hand, the discriminator's goal is to classify the images it has received as either real images or generated images from the generator. The GAN's objective function is a minimax game:

$$\min_{\theta_G} \max_{\theta_D} \mathbb{E}_{I^{HR} \sim p_{train}(I^{HR})}[\log D_{\theta_D}(I^{HR})] + \mathbb{E}_{I^{LR} \sim p_G(I^{LR})}[\log(1 - D_{\theta_D}(G_{\theta_G}(I^{LR})))]$$

where I^{HR} $p_{train}(I^{HR})$ are the input high resolution images, I^{LR} $p_G(I^{LR})$ are the input resolution images, G_{θ_G} is the output of the generator and D_{θ_D} is the output of the discriminator. We use the perceptual loss function for VGG based content losses introduced by Ledig et. al (7) which is a weighted sum of a content loss (l_X^{SR}) and an adversarial loss component ($10^{-3} l_{Gen}^{SR}$).

For the content loss, we use the VGG loss introduced by Ledig et. al (7) which is the euclidian distance between the feature representations of a reconstructed image $G_{\theta_G}(I^{LR})$ and the reference image I^{HR}:

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

where $W_{i,j}$ and $H_{i,j}$ describe the dimensions of the respective feature maps within the VGG19 network. The adversarial generative loss l_{Gen}^{SR} is defined based on the probabilities of the discriminator $D_{\theta_D}(G_{\theta_G}(I^{LR}))$ over all the training samples as:

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

$D_{\theta_D}(G_{\theta_G}(I^{LR}))$ is the probability that the restructured image $G_{\theta_G(I^{LR})}$ is a natural HR image. For better gradient behavior, we minimize $-logD_{\theta_D}(G_{\theta_G}(I^{LR}))$ instead of $log[1 - D_{\theta_D}(G_{\theta_G}(I^{LR}))]$.

5 Experiments, Results and Discussion

5.1 Hyperparameters

For image colorization, we used the AdamOptimizer (18) with a learning rate of 0.001, a beta1 of 0.9, a beta2 of 0.999, and epsilon=1e-08. We trained for approximately 41,000 iterations. We visually inspected the results every 1000 iterations. Initially the results were poor, but after approximately 30,000 iterations the results began to exhibit signs of overfitting. For the SRGAN, we used a learning rate of 10^{-4}, $10^(6)$ update iterations and the AdamOptimizer with $\beta_1 = 0.9$. We alternate updates to the generator and discriminator network similar to k = 1 as used by Goodfellow et. al (6). Our generator network has 16 identical residual blocks. During test time, we turn off our batch normalization layers (19) to obtain deterministic outputs.

5.2 Results

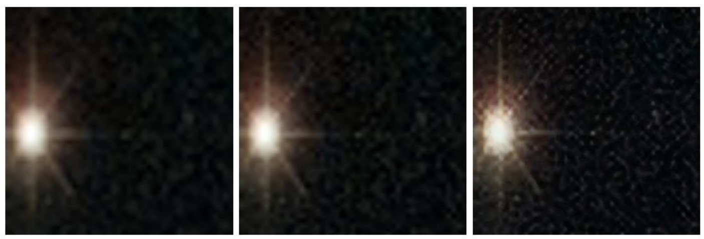
Our input images into the Image Colorization + SRGAN model were greyscale and of dimension (2040, 1356). The images produced from our image colorization model is of dimension (224, 224) and after 4x upscaling by our GAN, our output images are of dimension (896, 896). These dimensions are due to the variable, but typically small dimensions of raw Hubble images, and so while we wanted extraordinarily high-resolution images to pre-train the GAN with, we were most interested in being able to ultimately input the low resolution raw Hubble images, which dictated the size of the colourization network and subsequently the GAN.

For colorization, we use Mean Squared Error (MSE) to measure the average squared difference between the artificially colored images and the original RGB images. A larger MSE represents a lower quality of colorizing a grayscale image. For image super resolution, we use the Peak Signal to Noise Ratio (PSNR) to measure the quality of our GAN-produced reconstructed images. A larger PSNR represents a higher quality image. PNSR is calculated as $10 \log_{10} \frac{255^2}{\mathtt{MSE}}$. When calculating these metrics, we downsized the extremely high resolution original down to the size of the generated image. We calculated the Inception Score (20) to evaluate our GAN. A larger Inception Score (IS) indicates that the GAN-produced images are both salient and diverse. The IS is calculated using this formula: $IS(x) = exp(\mathbb{E}_x[KL(p(y|x)||p(y))])$

Model	Mean Squared Error	Peak Signal to Noise Ratio	Inception Score
Image Colorization + SRGAN	201.769	27.393	3.262 Std. Dev (0.327)
Bicubic Interpolation	180.271	38.092	-
Original Images	-	-	3.447 Std. Dev (0.206)

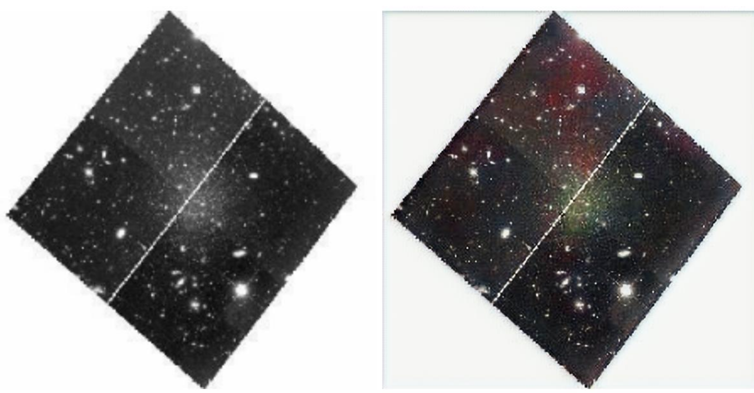
Qualitatively, we observe high-quality results, for colourization and super-resolution. There are undeniably flaws; colours tend toward the red and green, with almost no blues. Colours are also more washed out. The quantitative errors may actually underestimate the quality of the results, given that *all* colours are artificial, and what matters is much more the qualitative visuals of the result, rather than a true distance. The super-resolution aspect of the project also performed well; visually, it looks significantly sharper than the bi-cubic baseline. PSNR seems to be a misleading metric as it focuses on pixel-by-pixel differences instead of looking at the image as a whole; our SRGAN approach gives dramatically sharper images than the bi-cubic baseline, despite receiving a lower score.

The Inception Scores for our GAN-generated images versus the original images are very close to each other which means our GAN did produce images that are salient and diverse. Saliency is the concept where a human, on looking at the image, will be able to confidently determine what is in there. Diversity is the concept where a human, looking at a set of various images, will say that the set has lots of different objects.



Left to right: Original, bi-cubic baseline, clearly better SRGAN version

To demonstrate the utility of our system, we colorized and improved the resolution of the low-resolution, greyscale images from the Hubble Legacy Archive. We found that colourization and super-resolution both worked extremely well, producing noticeably higher-quality images. Though harder to distinguish in the images below, manual examination of generated images shows consistent improvement over raw images.



Left: Raw Hubble image. Right: Colourized, super-resolution image

6 Conclusion/Future Work

Our images could be directly studied by astronomers (21). An image stitching algorithm can be applied on our images to generate large-scale astronomical images for scientific study. Colourization can be improved with further experimentation with weighted loss functions. In particular, we can increase the loss for low saturations, a common problem we saw. We could also attempt to use a single GAN for both colourization and super-resolution. Our project demonstrates that automatic colourization and super-resolution produces images far better than the raw data collected by Hubble.

Further modifications that can be done to the SRGAN include the WGAN (22) which makes training more stable by forcing the discriminator to stay within the space of 1-Lipschitz functions. Gradient penalty (22) can also be added as an alternative to weight clipping in WGANs. Progressively Growing GANs (23) can also be added to the network in order to progressively grow the resolution of the generator and the discriminator by adding new layers during the training process.

7 Contributions and Acknowledgements

Both members of the team contributed equally, both to this report and on the technical front. Though of course there was overlap between both roles, Kai focused on the dataset scraping and preprocessing, setting up AWS, and building the colourization networks. Gao focused on the background research and implementing the GAN. We would like to thank Sarah Najmark, our project mentor, for her guidance. She went above and beyond to mentor us and answer our questions.

References

[1] S. P. Corp, "Creating hubble's technicolor universe."

[2] S. O. of Public Outreach, "Hubble image processors."

[3] R. Dahl, "Automatic colorization."

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[7] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[9] X. Yu and F. Porikli, "Ultra-resolving face images by discriminative generative networks," in *European conference on computer vision*, pp. 318–333, Springer, 2016.

[10] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[11] J. Bruna, P. Sprechmann, and Y. LeCun, "Super-resolution with deep convolutional sufficient statistics," *arXiv preprint arXiv:1511.05666*, 2015.

[12] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*, pp. 694–711, Springer, 2016.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[14] B. Dunbar, "Hubble's largest galaxy portrait offers a new high-definition view."

[15] C. Guo, "Automatic image colorization."

[16] H. Dong, A. Supratak, L. Mai, F. Liu, A. Oehmichen, S. Yu, and Y. Guo, "TensorLayer: A Versatile Library for Efficient Deep Learning Development," *ACM Multimedia*, 2017.

[17] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[20] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in neural information processing systems*, pp. 2234–2242, 2016.

[21] K. Schawinski, C. Zhang, H. Zhang, L. Fowler, and G. K. Santhanam, "Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit," *Monthly Notices of the Royal Astronomical Society: Letters*, vol. 467, no. 1, pp. L110–L114, 2017.

[22] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[23] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

Automatic image colourization based off of
`https://github.com/Armour/Automatic-Image-Colorization`

Image Super-Resolution using GANs based off of
`https://github.com/tensorlayer/srgan`

Inception Score based off of
`https://github.com/tsc2017/Inception-Score`

Our Github Repositories:
`https://github.com/KMarshland/image-colourization`
`https://github.com/gao-xian-peh/astronomy_super_resolution`