CS230

Skin Lesions Images Super-Resolution to Assist Diagnosis

Hsu-Hang (Eric) Yeh
Stanford University
ericyeh@stanford.edu

Joshua Vernon Tanner
Stanford University
jvtanner@stanford.edu

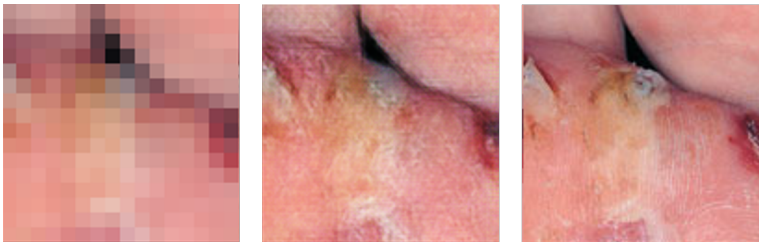

Figure 1: 16x16 LR image, 128x128 SR image, 128x128 original HR image

Abstract

Converting a low-resolution skin image to a high-resolution one is essential to making correct diagnosis in many clinical scenarios. Several different approaches have been attempted to tackle this task, but it is challenging to restore resolution while preserving precise edge details. We aim to build a deep-learning end-to-end model that utilizes edge information to solve this problem. We experimented with GAN-based methods that are using canny edge maps or gradient maps (edge-informed SISR, modified SPSR) and segmentation-based methods based on gradient maps(modified Unet, modified Unet++). We found edge-informed SISR that utilized canny edge maps achieved both the highest human rating scores (both dermatologists and non-dermatologists) and the highest image sharpness, but the modified SPSR which used gradient maps had the shortest feature vector distance from HR images at a scaling factor of 4. Edge-informed SISR was also able to generate sharp details at a scaling factor of 8. (shown in Figure 1.)

1 Introduction

Dermatologists rely on high-resolution (HR) pictures to make diagnoses when direct observation is impossible. This is especially common in telemedicine, medical consultations, and regular clinics. However, the resolution of these pictures is often compromised by technical limitations or poor photography skills. In order to improve the quality of these pictures, we aim to build a deep-learning model that takes low-resolution (LR) skin lesion images as inputs and reconstructs photo-realistic super-resolution (SR) images.
Because an SR image represents just one of many unique possible interpretations of a single LR image, determining the most accurate SR image has proven a challenging task. Non-learning based approaches have been attempted, but these methods have been limited by poor generalizability and inaccuracy. In contrast, deep learning-based SR reconstruction has provided promising results.[1] Recently GAN has gained popularity and has become the heart of many state-of-the-art models. The visual quality of its generated images is indeed impressive, yet it too has demonstrated limitations. GAN struggles to preserve edge details, as straight lines are frequently distorted and wavy. To overcome this problem, several attempts on preserving edge information have been explored. It is primarily upon this issue of edge integrity preservation which drives our task at hand.

2 Related work

Several modifications of GAN-based methods are designed to send edge information into the model to preserve edge details. The edge-informed SISR [2] seeks to ameliorate this issue by utilizing a two-stage adversarial model consisting of an canny edge-extraction step followed by an image completion step. The canny edge map serves as a guide for image completion to avoid edge distortion. On the other hand, structure-preserving SR (SPSR) [3] aims to preserve edge information by using gradient maps as guidance. A gradient

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

map was obtained by calculating pixel difference between neighboring pixels in both directions, and deriving the magnitude of the gradient at each point:

$$\mathcal{G}(x,y) = \sqrt{\mathcal{G}_x^2 + \mathcal{G}_y^2},\ where\ \mathcal{G}_x(x,y) = I(x+1,y) - I(x-1,y),\ \mathcal{G}_y(x,y) = I(x,y+1) - I(x,y-1)$$

Instead of using separate GANs, it combines both gradient-extraction and image completion in a single GAN by placing them in different branches. The information from the image branch will be sent to gradient branch to improve the quality of gradient reconstruction. Results from both branches are fused and passed in a final convolutional layer to generate the final results.
Segmentation-based methods have also been explored as a means to solve SISR tasks, and has been shown by some researchers to be quite effective in preserving edges. [4, 5] Unet, which was famous for segmentation in biomedical domain, was primarily used in these studies. In [4] the authors added two up-sampling layers at the beginning and the end of the Unet architecture, with skip connections between them. In each convolutional block, they removed the batch normalization and only kept only one of the convolutional layers. To avoid edge distortion, they incorporated a gradient loss term in the loss function.

3 Dataset and Features

The images for this project were drawn from Atlas of Clinical Dermatology, a dataset built by Danish dermatologist Dr. Niels K. Veien in a private dermatology practice. The pictures were downloaded from http://www.danderm-pdv.is.kkh.dk/atlas/index.html. We excluded pictures taken with microscopes or dermoscopes since the need for higher resolution is minimal. We also excluded any picture not depicting skin lesions because they rarely need SR reconstruction in clinical practice. There are a total of 3425 images included with ratios similar to the frequencies of various disease entities seen in dermatology clinics. We observed a great variation in picture resolutions, ranging from 420x298 to 1333x2000. To address this issue, we randomly cropped the images in the training dataset to a fixed size of 128x128 before feeding them into the model. This cropping also facilitated the training process due to lower computational cost per image. Each image contained a small watermark "D@nderm" at the corner for protecting copyright. We chose to keep this watermark due to its relatively small size and ubiquitous distribution.
We randomly chose 570 images for the development set and another 570 images for the test set. For data augmentation, we applied 90 degrees counterclockwise rotation on the remaining 2285 pictures, which gave us a training dataset of size 4570. The ratio of train/dev/test was roughly 8:1:1. To generate pairs of ground-truth HR images and LR images, each 128x128 image was downsampled by the scaling factor using a bicubic interpolation.

4 Models

4.1 Edge-informed SISR

We built our first model upon the official implementation of edge-informed SISR by Kamyar Nazeri. [2] The model had one GAN for generating HR canny edge maps followed by another for SR images. The inputs of the first GAN were LR images and LR canny edge maps which were obtained by applying canny edge detector on LR images. Before fed into the model, both LR images and edge maps underwent interpolation to the target size using nearest-neighbor algorithm. An edge discriminator was used to discriminate between real HR edge maps and generated HR edge maps. The generator was optimized by both the adversarial loss and the feature matching loss, which was defined as L1 loss between feature maps of the fake edge map and the real edge map at each layer of the discriminator:

$$\mathcal{L}_{G_1} = \lambda_{G_1}\mathcal{L}_{adv,G_1} + \lambda_{FM}\mathcal{L}_{FM}$$

$$\mathcal{L}_{FM} = \mathbb{E}\left[\sum_i \frac{1}{N_i}\|D^{(i)}_{edge}(E_{gt}) - D^{(i)}_{edge}(E_{pred})\|_1\right]$$

The inputs of the second GAN were the generated HR edge map and the "dilated" LR image, which was created by padding zeros to the the right and the bottom of each LR image pixel to inflate the image to the HR size. We utilized another discriminator to test the realness of generated SR images. In addition to adversarial losses and pixel-wise L1 losses, we used a pre-trained VGG19 network to extract feature maps and calculated both content losses and style losses for the second generator. The content loss is defined as the sum of L1 distances between intermediate feature maps, and the style loss is the sum of L1 distances between Gram matrices of those feature maps.

$$\mathcal{L}_{G_2} = \lambda_{L1}\mathcal{L}_{L1} + \lambda_{G_2}\mathcal{L}_{adv,G_2} + \lambda_{content}\mathcal{L}_{content} + \lambda_{style}\mathcal{L}_{style}$$

$$\mathcal{L}_{content} = \mathbb{E}\left[\sum_i \frac{1}{N_i}\|\phi_i(I_{gt}) - \phi_i(I_{pred})\|_1\right],\ \mathcal{L}_{style} = \mathbb{E}\left[\sum_i \|G^{\phi}_i(I_{gt}) - G^{\phi}_i(I_{pred})\|_1\right]$$

4.2 Modified SPSR

The 1st GAN of edge-informed SISR failed to generate precise gradient maps in our experiments. To solve this problem, we were inspired by the work of Cheng Ma et al.[3] and decided to incorporate both the gradient generator and the SR generator into a single network. We built the model from scratch, using the generators in edge-informed SISR in both branches. We chose to send output

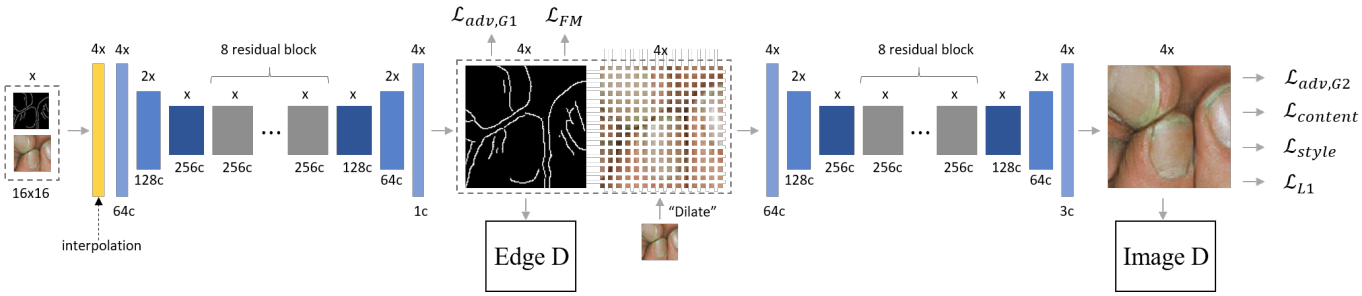

Figure 2: Edge-informed SISR architecture, D:discriminator, FM:feature matching, adv:adversarial, G:generator, c:channels

from the second encoder and the second decoder in the SR branch to the gradient generator for a more accurate gradient reconstruction. The output from the SR branch and the gradient branch were subsequently fused together and passed into a final 3x3 convolutional layer to generate the final SR image. We also made modifications in the perceptual loss, where we used both content and style losses instead of just the content losses. The total loss was composed of adversarial loss and L1 losses of both SR images and gradient maps, content loss and style losses extracted from VGG19 network(Br :from gradient branch, SR :from SR branch):

$$\mathcal{L}_G = \lambda^{SR}_{adv,G}\mathcal{L}^{SR}_{adv,G} + \lambda^{SR}_{content}\mathcal{L}^{SR}_{content} + \lambda^{SR}_{style}\mathcal{L}^{SR}_{style} + \lambda^{SR}_{L1}\mathcal{L}^{SR}_{L1} + \lambda^{SR,Grad}_{adv,G}\mathcal{L}^{SR,Grad}_{adv,G} + \lambda^{SR,Grad}_{L1}\mathcal{L}^{SR,Grad}_{L1} + \lambda^{Br,Grad}_{L1}\mathcal{L}^{Br,Grad}_{L1}$$

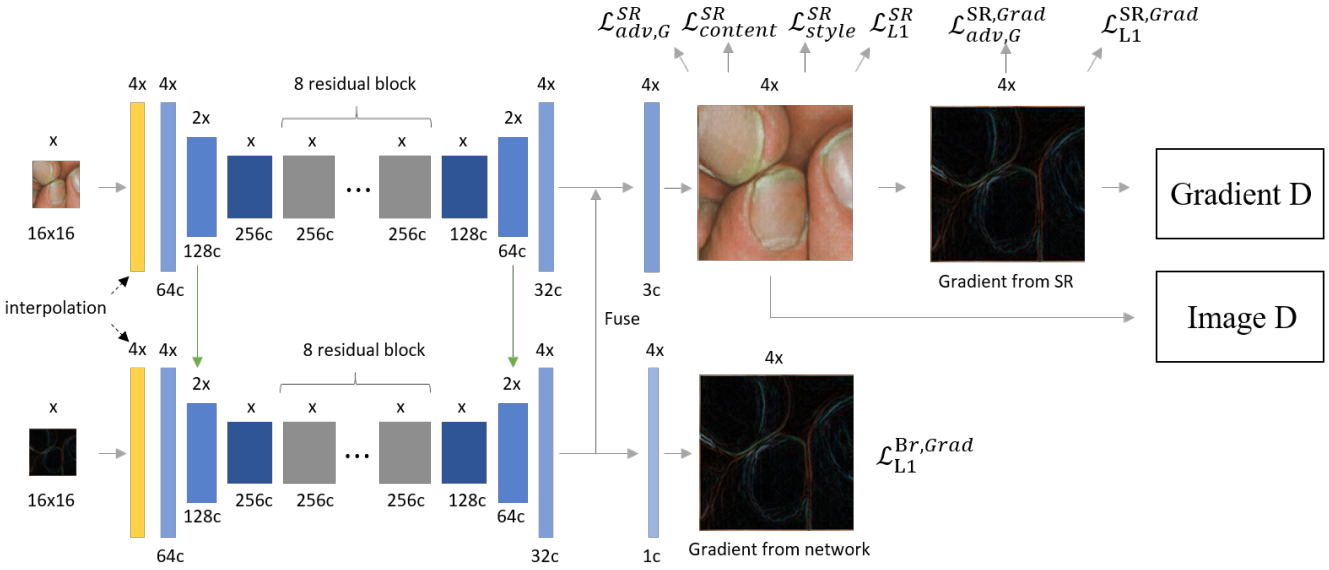

Figure 3: Modified SPSR architecture, D:discriminator, adv:adversarial, G:generator, c:channels

4.3 Modified Unet

We built our modified Unet model from scratch following the structure proposed by Zhengyang Lu et al. [4]Two upsampling layers were appended to the beginning and the end of the Unet. There were four downsampling steps in the encoder path. Each convolutional block contained one convolution layer followed by ReLu activation. The model was then optimized by incorporating losses of both the image and the gradient map (both pixel-wise L1 and MSE were experimented): $\mathcal{L}_{total} = \lambda_{pixel}\mathcal{L}_{pixel} + \lambda_{gradient}\mathcal{L}_{gradient}$

4.4 Modified Unet++

We also built modified Unet++ from scratch. The regular Unet was replaced with a nested Unet. We chose Unet++ for its proven superior results in biomedical image segmentation. [6] We resumed the number of convolutional layers in each convolutional block to 2 for better performance. Because the model now had 6 more convolutional blocks, we had to reduce the channel numbers to half of the original size.

5 Metrics

Peak Signal-to-Noise Ratio (PSNR), Structural similarity (SSIM) were conventionally used in SISR tasks. By convention all metrics were calculated in the Y channel of YCbCr color space. We converted images to YCbCr in the same way as MATLAB rgb2ycbcr

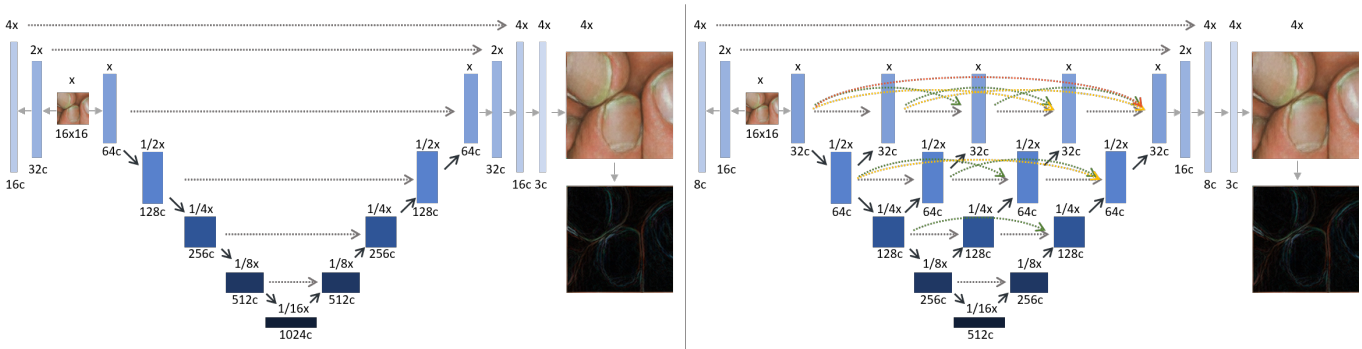

Figure 4: modified Unet/Unet++ architecture, c:channels

method did.[7] Because these metrics often failed to correlate with human perceptual qualities, we also included a human-rating scoring metric. The 570 test images were randomly separate into 6 groups, each for head-to-head comparison between 2 of the models we used. For each comparison, a model was given a score of +1 when an observer judged its SR image to be of superior visual quality than another's. The final score was normalized by dividing by the total number of comparisons. This was a blind process as pictures were randomly selected with model names withheld. Non-dermatologist ratings were accomplished by our team members. We also invited a dermatologist at a medical center in Sounthern Taiwan to perform the dermatologist rating.
To establish more objective metrics, we used the variance of the Laplacian, which was previously used as a focus measure, to estimate how many edges exist in an image.[8] We believe this could be an approximation of how well the edge information is preserved. On the other hand, to estimate the content fidelity of SR images, we utilized VGG16 and ResNet50 pretrained on ImageNet to extract the feature vectors and calculated the Euclidean distance from the original HR images to the SR images in the high-dimensional vector space.

6 Experiments

The experiments were conducted on an Amazon E2 instances using 1 NVIDIA K80 GPU with 12GB memory. All models were trained in a similar fashion. We used a batch size of 8 to fit the GPU memory. The models were optimized by an Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$.(for modified SPSR $\beta_2 = 0.999$) We trained the models at a learning rate of 0.001 until the validation loss reached plateau. We then decreased the learning rate to 0.0001 until convergence.
In edge-informed SISR, we experimented with different weights to determine which loss term was the most crucial in reconstructing realistic skin images. The results showed that the original author's weights gave the best results.($\lambda_{G_1} : 1,\ \lambda_{FM} : 10,\ \lambda_{L1} : 1,\ \lambda_{G_2} : 1,\ \lambda_{content} : 0.1,\ \lambda_{style} : 250$) We also tried different sigma values of the canny edge detector and found 2 was enough for capturing details without producing too much noise. To compare between canny edge maps and gradient maps, we built a gradient-informed SISR by substituting the edge-extracting GAN with a gradient-generating GAN. Unfortunately, the gradient map could not be learned efficiently with our architecture, and the resulting SR images were blurry.
We performed similar weights tuning in modified SPSR. Instead of giving more weights to the style loss, we found that equal ratios in the content and the style loss gave the best results.($\lambda^{SR}_{content} : 1,\ \lambda^{SR}_{style} : 1$) For other weights, the ratios in the original paper performed the best. ($\lambda^{SR}_{adv,G} : 0.005,\ \lambda^{SR}_{L1} : 0.01,\ \lambda^{SR,Grad}_{adv,G} : 0.005,\ \lambda^{SR,Grad}_{L1} : 0.01,\ \lambda^{Br,Grad}_{L1} : 0.5.$)
For both modified Unet and Unet++, we weren't able to get clear SR images. To force the model to generate more realistic pictures, we added content loss and style loss as defined previously but only observed minimal improvement. We found using L1 loss of both images and gradient maps gave more sharp results than using an MSE loss. Given that the gradient map representing neighboring pixel differences might not contain enough information, we switched to Sobel operator for calculating the gradients. However, this only brought about negligible improvement.

7 Results

Uncurated SR images from different models were presented in Figure 5. Notice the sharper edges in edge-SISR and SPSR compared to Unet and Unet++. Both human rating and objective metrics were summarized in Table 1. PSNR failed to correlate with human perceptual quality as expected. Although SSIM reportedly agreed more with human perception [9], it failed to correlate with human ratings in our data. Non-dermatologist ratings were consistent with dermatologist ratings and could serve as a proxy of diagnostic values.
The edge-informed SISR was successful in both human ratings and vatiance of the Laplacian, while SPSR fell behind. Because both were using the same generators, one possible reason for the failure of SPSR was the shallowness of CNN after branch fusion, which limits the ability to incorporate the gradient information. We found that edge-informed SISR was prone to generate artificial lines if many edges existed in original pictures. As shown in Figure 5(b), the fingerprints were mixed with lines vertical to its natural direction.

Our modified SPSR outperformed edge-SISR in this regard. Notice the direction was well preserved in modified SPSR model. This was also consistent with the fact that SPSR results had the shortest distances from HR images in the high-dimensional feature vector space. This might indicate that the gradient maps have more potential in restoring realistic edges. The color, however, was not well restored by SPSR. We believe this could also be solved by using deeper CNN after branch fusion.
Both modified Unet and Unet++ failed to reproduce sharp image details, suggesting that intermediate convolutional layers were not helpful. The problem might result from the absence of edge information in model inputs. We also frequently observed randomly distributed color spots in the generated images. This phenomenon existed regardless of how we obtained the gradient maps. This might originate from the large weight of the gradient loss.
Overall, edge-informed SISR performed the best with our data. We went ahead and trained the model to do a 8X super-resolution. Again, the model failed on images with excessive edges like hairs or wrinkles. But it could produce realistic results for images with moderate amount of edges. One of the successful results are shwon in Figure 1.

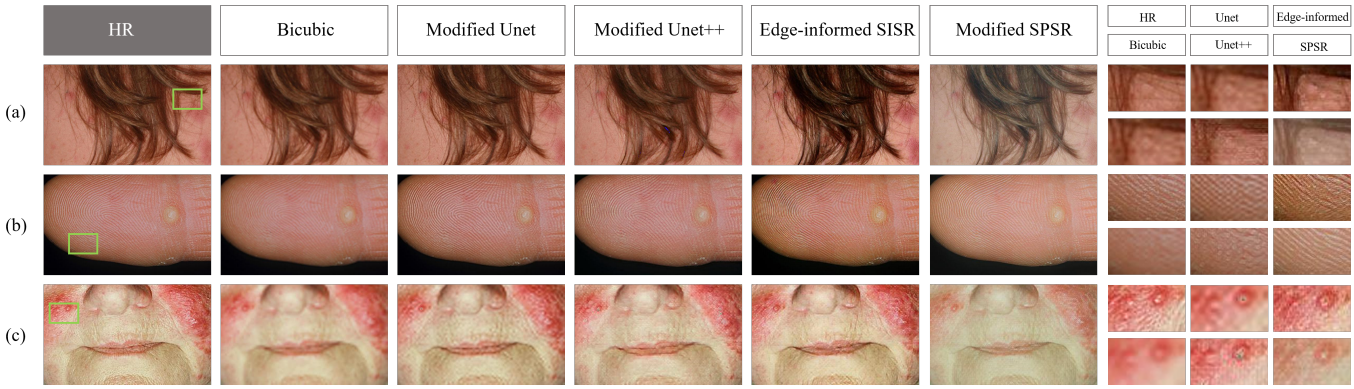

Figure 5: 3 uncurated SR images at 4X. Green box area is magnified on the right.

	PSNR	SSIM	Non-dermatologist Rating	Dermatologist Rating	Variance of the Laplacian (HR=510)	VGG16 Feature Vector Distance from HR	ResNet50 Feature Vector Distance from HR
Bicubic	31.43	0.79	NA	NA	9	1315	659
Edge-informed SISR	28.13	0.75	0.80	0.88	332	1224	594
Modified SPSR	26.25	0.78	0.73	0.71	130	1021	554
Modified Unet	30.45	0.80	0.21	0.19	135	1470	697
Modified Unet++	29.48	0.72	0.24	0.22	237	1535	714

Table 1: Subjective and objective metrics at 4X. Green indicates the best and orange indicates the second best.

8 Conclusion/Future Work

GAN-based methods produced more visually realistic skin images at 4X, and the edge-informed SISR gave the most promising results. To eliminate the line artifacts of edge-informed SISR, we might need to construct novel loss terms to penalize random directions of local edges, since most edges on the skin should have similar directions locally. Our modified SPSR generated results closest to original images. To improve the colors and details of SPSR-generated images, we could build a deeper CNN after the branch fusion.

9 Contributions

Hsu-Hang (Eric) Yeh conceived and planned the experiments, downloaded the dataset, built and trained the models, calculated the metrics, and drew illustrations and wrote drafts for reports. Joshua Vernon Tanner contributed to literature reviews, preparing the dataset, rating results, drafting and revising the reports, and debugging the models.

10 Acknowledgements

We would like to thank Pranav Rajpurkar for his valuable feedbacks on our projects. We also thank Ting-Jung, Hsu for performing the dermatologist ratings for us.

References

[1] Wenming Yang et al. "Deep Learning for Single Image Super-Resolution: A Brief Review". In: *CoRR* abs/1808.03344 (2018). arXiv: 1808.03344. URL: http://arxiv.org/abs/1808.03344.

[2] K. Nazeri, H. Thasarathan, and M. Ebrahimi. "Edge-Informed Single Image Super-Resolution". In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, pp. 3275–3284.

[3] Cheng Ma et al. "Structure-Preserving Super Resolution with Gradient Guidance". In: *arXiv preprint arXiv:2003.13081* (2020).

[4] Zhengyang Lu and Ying Chen. "Single Image Super Resolution based on a Modified U-net with Mixed Gradient Loss". In: *arXiv preprint arXiv:1911.09428* (2019).

[5] Xiaodan Hu et al. "RUNet: A Robust UNet Architecture for Image Super-Resolution". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2019.

[6] Zongwei Zhou et al. "Unet++: A nested u-net architecture for medical image segmentation". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer, 2018, pp. 3–11.

[7] xinntao. *Color conversion in SR*. URL: https://github.com/xinntao/BasicSR/wiki/Color-conversion-in-SR.

[8] Said Pertuz, Domenec Puig, and Miguel Ángel García. "Analysis of focus measure operators for shape-from-focus". In: *Pattern Recognit.* 46 (2013), pp. 1415–1432.

[9] Umme Sara, Morium Akter, and Mohammad Shorif Uddin. "Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study". In: *Journal of Computer and Communications* 7.3 (2019), pp. 8–18.