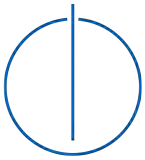
SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Learning-Based Ultrasound Image Super-Resolution

Ozan Karaali



SCHOOL OF COMPUTATION,
INFORMATION AND TECHNOLOGY
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Learning-Based Ultrasound Image Super-Resolution

Lernbasierte Ultraschallbild-Super-Resolution

Author:	Ozan Karaali
Supervisor:	Prof. Dr. Nassir Navab
Advisor:	Dr. Shahrooz Faghih Roohi
Submission Date:	15.05.2023

I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, 15.05.2023 Ozan Karaali

Acknowledgments

I would like to express my deepest gratitude to Prof. Dr. Nassir Navab and Dr. Shahrooz Faghih Roohi for supervising and advising me throughout this thesis with their guidance and support and for having me as their thesis student. I am thankful to Chair for Computer Aided Medical Procedures and the members for providing me with resources and giving me feedback on this thesis. I am grateful for their encouragement, constructive criticism, support, and investment in my academic journey and I have learned a lot from their feedback and guidance. I am also grateful to my lecturers, research assistants and classmates which impacted me along the way. Thank you for your time, attention, and dedication to my success.

I want to thank all the authors who have contributed to my research project with their works. Their insights have been invaluable in shaping the outcomes of this study. Their work helped me to understand the subject better and accelerated my experiments. I am grateful for their contributions to the field of computer vision and their willingness to share their knowledge with the community.

In addition, I want to thank my mother and father for their unconditional love and support. Their constant encouragement and belief in me have been a driving force. Without their sacrifices, I would not have been able to pursue my education and achieve my academic goals. And, I also owe a debt of gratitude to my friends for their support. They have been there for me every step of the way and offered encouragement when I needed them most.

To all those who have supported me, thank you from the bottom of my heart. Your contributions have made this thesis journey possible, and I am forever grateful.

Ozan Karaali

Abstract

LEARNING BASED ULTRASOUND IMAGE SUPER-RESOLUTION

Ozan Karaali
Supervisor: Prof. Dr. Nassir Navab

B-mode ultrasound imaging is commonly used by physicians and clinicians for diagnosing and treatment purposes by visualizing and quantifying anatomical structures in an easier environment and under a relatively cheaper budget when compared to other imaging methods. Due to the physical constraints of ultrasound imaging devices and medium (tissue) limits, the inherent features of ultrasound and the quality of ultrasound imaging is never optimum, particularly limited by the spatial resolution. To overcome these limitations, image super-resolution (SR) technology may aid in clinical medical diagnosis and therapy by improving ultrasonic imaging quality and increasing illness diagnostic accuracy. However, the resolution degradation process of ultrasonic imaging in actual scenarios is unpredictable due to the changes in sensor equipment or transmission medium. In this thesis, several image super-resolution (SR) technologies, such as previous works in ultrasound B-mode image SR methods and then natural image SR methods and degradation aware SR methods, will be investigated, and an upgrade for image super-resolution (SR) technology by utilizing deblurring networks to fine-tune the degradation on the ultrasound images be introduced to be employed with B-mode ultrasound imaging, additionally discovering ways to utilize SOTA natural image SR methods in ultrasound images will be investigated under different degradation scenarios.

Keywords: Ultrasound Image, Super-Resolution, Deep Learning, Deblurring, Denoising, Image Enhancement

Contents

1 Introduction

Ultrasound imaging has become an increasingly popular diagnostic tool due to its non-invasive and real-time capabilities. However, the quality of ultrasound images is often limited by the physical constraints of the imaging system, such as the limited resolution and the presence of noise. The image super resolution (SR) technology can improve the quality of ultrasound images, which helps to overcome these limitations.

1.1 Ultrasound Imaging

Ultrasound imaging is a non-invasive imaging technology that produces images of the body's internal structures by using high-frequency sound waves. The ultrasound imaging system comprises a transducer, a signal processing unit, and a display unit. [RL17] The transducer generates and receives the ultrasound waves, which are then processed by the signal processing unit to create the final ultrasound image.

Ultrasound waves are high-frequency sound waves that exceed the human hearing range. In ultrasound imaging, typically sound waves with a frequency range of 2-15 MHz are used, but not limited to that, for some specialized imaging applications, frequencies as high as 60 MHz, also investigated [RL17]. The speed of sound determines the wavelength of the ultrasound waves in the medium and the frequency of the waves, according to the relationship

$$\lambda = \frac{c}{f}$$

where lambda is the wavelength, c is the speed of sound, and f is the frequency of the waves [RL17].

As shown in Figure 1.1, the transducer generates the ultrasound waves and propagates them through the body. When the ultrasound waves encounter an interface between two different media with different acoustic properties, wave components are reflected in the transducer, while others are transmitted further into the body. Then, the reflected waves are used to create the ultrasound image [RL17].

There are fundamentally four imaging modes for ultrasound imaging [RL17]:

- amplitude-mode (A-mode) ultrasound records the position and strength of a reflecting structure.
- motion-mode (M-mode) ultrasound shows echo amplitude and the position of moving reflectors.

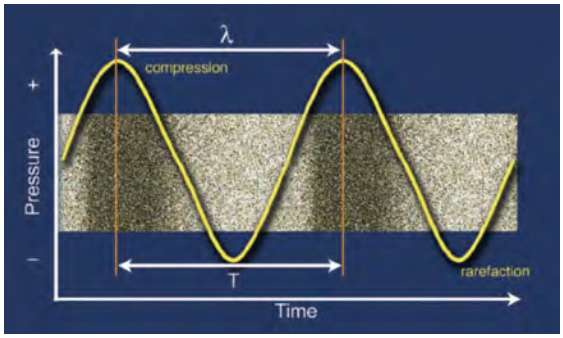


Figure 1.1: A sound wave is a series of alternating pressure waves producing compressions and rarefactions on the conducting medium. [RL17]

- brightness-mode (B-mode) ultrasound shows 2D real-time, grayscale images where brightness indicates reflecting signals of differing amplitude.
- Doppler mode ultrasound uses the Doppler effect to measure and visualize blood flow.

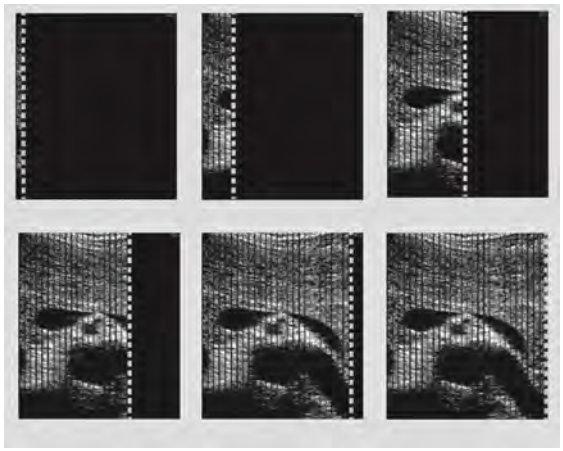


Figure 1.2: Ultrasound pulses delivered down a series of successive scan lines combine to create a 2D real-time image. Each scan line adds to the image, creating a 2D representation of the echos from the item being scanned. [RL17]

This thesis focuses on the B-mode ultrasound imaging, which is the most commonly used imaging mode in clinical practice. The B-mode ultrasound imaging is a 2D real-time, grayscale imaging mode where the brightness of the image indicates the strength of the reflected signals built by a series of successive scan-lines in an array as shown in Figure 1.2. The B-mode ultrasound imaging is used to visualize the internal structures of the body, such as the heart, blood vessels, and the liver. The B-mode ultrasound imaging is also used to diagnose diseases and monitor their progression of them, such

as cancer, heart disease, and liver disease [RL17].

In the B-mode ultrasound imaging, the image is affected by various factors such as diffraction, attenuation, and aberrations, which cause blurring, reduced resolution, and low quality of the image. To model this blurring effect, in ultrasound imaging, a concept called the point spread function (PSF) is used. The PSF is a mathematical function that describes how a point source of ultrasound waves is spread out in the image due to the blurring effect of the imaging system. The PSF is important because it determines the resolution and quality of the final image. A sharper PSF results in a higher resolution image, while a broader PSF results in a lower resolution image.

The PSF is typically assumed to be a Gaussian function in ultrasound imaging due to its simplicity and effectiveness in modeling the blurring effect. The Gaussian PSF assumes that the blurring effect is symmetric and smoothly varying, which is a reasonable assumption for many ultrasound imaging applications [Zha+15].

1.2 Image Super-Resolution

Image SR is a process of enhancing the resolution of an image beyond what is physically or optically possible with the imaging system to improve the image's visual quality. The SR technique involves the reconstruction of a high resolution (HR) image from a low resolution (LR) image. To do that reconstruction, a function f should be realized to map the LR image I_{LR} to the HR image I_{HR}, where $I_{SR} = f(I_{LR})$, which I_{SR} should be as close as possible to I_{HR}.

B-mode ultrasound imaging displays 2D anatomical sections of human tissues as a grayscale image. High-resolution ultrasound images are helpful in observing the shape and contour to judge if there is a lesion on the tissue. However, the resolution of the ultrasound images is limited by the physical constraints of the imaging system; due to the diffraction, attenuation, and aberrations, it is difficult to obtain high-resolution ultrasound images, especially for deep tissues.

Where it is not feasible to obtain high-resolution ultrasound images, the SR technique can be employed to improve the quality of the ultrasound images.

The general resolution degradation model for the B-mode ultrasound imaging can be shown as follows [Liu+22]:

$$y = (x * k) \downarrow_s + n \tag{1.1}$$

where x is the high-resolution image and y is the low-resolution image. k is the point spread function (PSF), which is modeled as Gaussian blur kernel convolved on x, $\downarrow_s$ is the downsampling operator with the factor of s, and n is the noise. Since B-mode ultrasound imaging is similar to natural images, the general resolution degradation model can also be applied to B-mode ultrasound imaging. In addition to the general

resolution degradation model, ultrasonic imaging involves axial and lateral spatial resolution degradation. Even though the axial and lateral resolution degradations in ultrasound imaging are different, for convenience, they can be modeled with the uniform Gaussian blur kernel and the same downsampling operations for both dimensions as the general resolution degradation model since they will all lead to the degradation of the image quality [Liu+22]. Finally, authors of the "Progressive Residual Learning with Memory Upgrade for Ultrasound Image Blind Super-resolution" [Liu+22] state that the end-to-end deep SR methods based on bicubic downsampling are not applicable because the degraded blur kernel is generally unknown in actual ultrasound imaging process. Therefore Blind SR is more suitable for ultrasound imaging due to the unknown blur kernel.

There exist two different approaches for ultrasound super-resolution; the first one is called front-end mode SR, which aims to improve the image during the imaging process by changing the imaging conditions or the equipment, such as beam-forming or increasing the frequency while reducing the diffusion angle. In contrast, the second one is named back-end (soft) mode, which post-processes the obtained ultrasound image to improve its resolution [Liu+22]. This thesis aims to investigate the effectiveness of existing deep learning approaches for ultrasound super-resolution by applying to B-mode images to create soft mode super-resolution and compare their performance in terms of image quality and robustness to noise, then propose different approaches to improve the performance of existing deep learning approaches for ultrasound super-resolution. The results of this study might provide valuable insights for improving the resolution and quality of ultrasound images, which could ultimately enhance the accuracy and reliability of ultrasound-based diagnosis and treatment.

2 Related Work

Super-resolution has become a hot topic in image processing in recent years, with applications in surveillance, remote sensing, and medical imaging, among other fields. In order to improve the resolution and quality of low-resolution photos, super-resolution creates high-resolution versions of the same image. This reconstruction is accomplished using various methods, including interpolation, reconstruction, and deep learning-based methods.

In this section, the recent advancements in super-resolution techniques, focusing on deep learning-based approaches, will be reviewed. The section is divided into two parts: single image super-resolution and blind super-resolution. The first part will discuss the various deep learning-based approaches for single-image super-resolution. Then, in the second part, the various deep learning-based approaches for blind super-resolution will be discussed.

2.1 Super-Resolution

Like the natural image SR, earliest approaches such as Deep Convolutional Neural Network for Ultrasound Super Resolution (DECUSR) assumed image resolution degradation as bicubic sampling. Therefore, this section discusses different deep learning-based methods to reconstruct SR images with known blur kernel.

2.1.1 Interpolation

Interpolation is the process of estimating the values of a function at points between the known data points. In the context of super-resolution, interpolation is used to estimate the values of the high-resolution image at points between the known data points of the low-resolution image. In the Figure 2.1, the $4 * 4$ image is interpolated to a $16 * 16$ image using different interpolation algorithms. The interpolation algorithms are described in the following subsections.

Nearest Neigbor

This method is the most straightforward interpolation technique, where the new pixel value is estimated by replicating the value of the nearest neighboring pixel in the original image. This method is fast to compute and produces blocky results. Therefore it can result in jagged edges and aliasing artifacts.

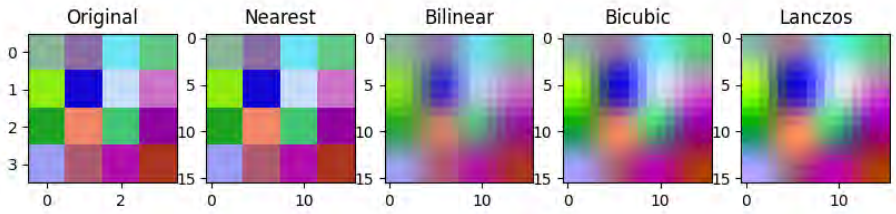


Figure 2.1: Different types of interpolation algorithms rendering upscaled images of the original $4 * 4$ image

Bilinear Interpolation

This method is an extension of the nearest neighbor method. It uses the four nearest neighboring pixels to estimate the value of the new pixel. This method produces smoother results than the nearest neighbor method but still suffers from aliasing artifacts.

Bicubic Interpolation

This method is an extension of the bilinear interpolation method. It uses the 16 nearest neighboring pixels ($4 * 4$ kernel) to estimate the value of the new pixel. This method produces smoother and more visually appealing results than the bilinear interpolation and nearest neighbor, but it is also more computationally expensive.

Lanczos Interpolation

Lanczos interpolation is a high-quality interpolation method that uses a windowed sinc function to estimate the value of the new pixel based on the values of neighboring pixels. This method produces the best results among the interpolation methods and is the most computationally expensive.

2.1.2 Natural Image Super-Resolution Methods

This section will review the various deep learning-based approaches for natural image super-resolution since B-mode ultrasound images are similar to natural images by containing three channels of RGB data used to display the grayscale image of anatomical sections and tissues in 2D.

Enhanced Deep Residual Networks

Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR) [Lim+17] architecture is one of the SR methods developed after the introduction of convolutional neural network (CNN). The architecture is based on the ResNet50 architecture [He+15]. The EDSR architecture is shown in Figure 2.2. The architecture comprises a feature extraction module, a residual block, and a reconstruction module. The feature extraction module is a CNN extracting features from the low-resolution image. The residual block is a CNN that is repeated multiple times to increase the depth of the network. The reconstruction module is a CNN that reconstructs the high-resolution image from the features extracted by the feature extraction module. The EDSR [Lim+17] architecture is trained using the L1 loss function; meanwhile, SRResNet [Led+16] used the L2 loss function [Led+16].

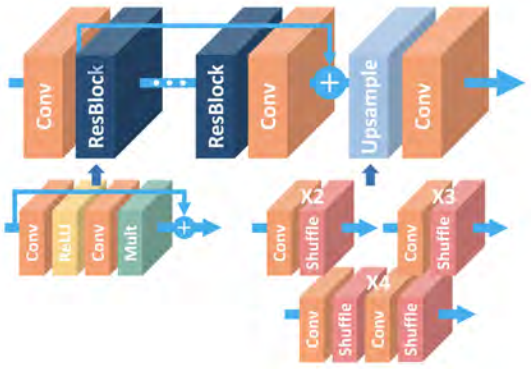


Figure 2.2: Model architecture of EDSR [Lim+17]

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

Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network (SRGAN) [Led+16] is a SR method that uses a generative adversarial network (GAN) to generate high-resolution images from low-resolution images. The SRGAN architecture is shown in Figure 2.3. The architecture consists of a generator and a discriminator. The generator is a CNN that generates high-resolution images from low-resolution images, which is based on SRResNet. With that part, it is similar to the EDSR architecture. In addition, what makes it different from EDSR and SRResNet is the discriminator part, which is a CNN that classifies the generated images as real or fake. Utilizing a GAN to generate high-resolution images from low-resolution images allows the generator to learn the characteristics of the high-resolution images instead of directly optimizing the loss function. The SRGAN architecture is trained using the perceptual loss function

and adversarial loss function [Led+16].

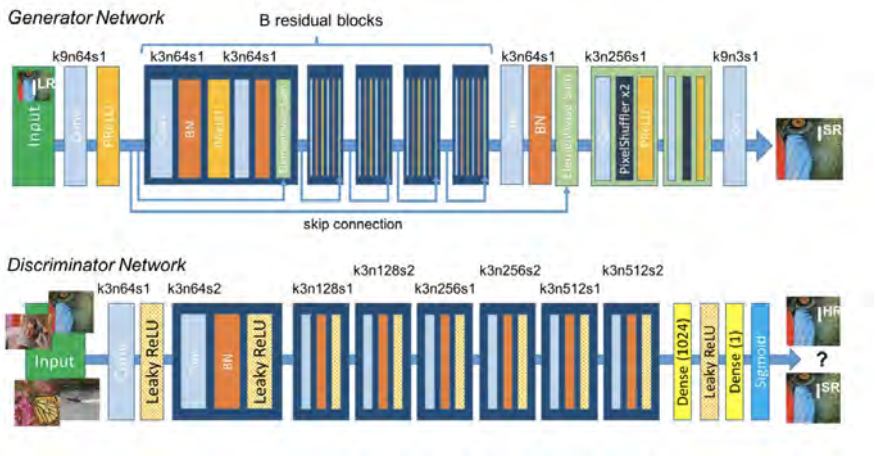


Figure 2.3: Model architecture of SRGAN [Led+16]

In addition, an extension to the SRGAN, ESRGAN tries to extend via utilizing Residual in Residual Dense Block (RRDB) [Wan+18] in the SRResNet-based architecture such as shown in Figure 2.4 and a mixture of context, perceptual and adversarial losses, so the total loss of the generator is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{perceptual}} + \alpha \mathcal{L}_{\text{G}} + \eta \mathcal{L}_{1} \tag{2.1}$$

where $\mathcal{L}_{\text{perceptual}}$ is the perceptual loss from VGG19 [SZ14], α is the weight of the GAN loss, η is the weight of the L1 loss, and $\mathcal{L}_{\text{G}}$ is the GAN loss.

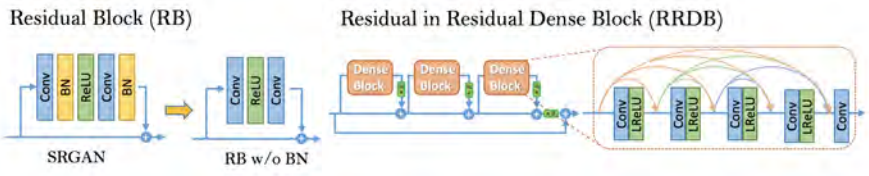


Figure 2.4: Model architecture of RRDB [Wan+18]

Swin Transformer

After the long domination of CNN based models in computer vision area in years, especially in natural language processing (NLP), Transformer [Vas+17] has emerged due to its use of attention mechanism to model long-range dependencies in the data. In later stages, the exact Transformer mechanism has adapted to the computer vision area as Vision Transformers [Dos+20] first, then to handle the difference between

language and computer vision domains such as the scale of visual entities and the high resolution of pixels in images compared to words in the text the Shifted Window Transformer (Swin) is proposed. The difference between a Vision Transformer and Swin transformer can be seen in Figure 2.5 and the whole architecture of Swin Transformer can be seen in Figure 2.6 According to Swin Transformer's research, by limiting self-attention computation to non-overlapping local windows and allowing for cross-window connection, shifted windowing approach increased efficiency [Liu+21c].

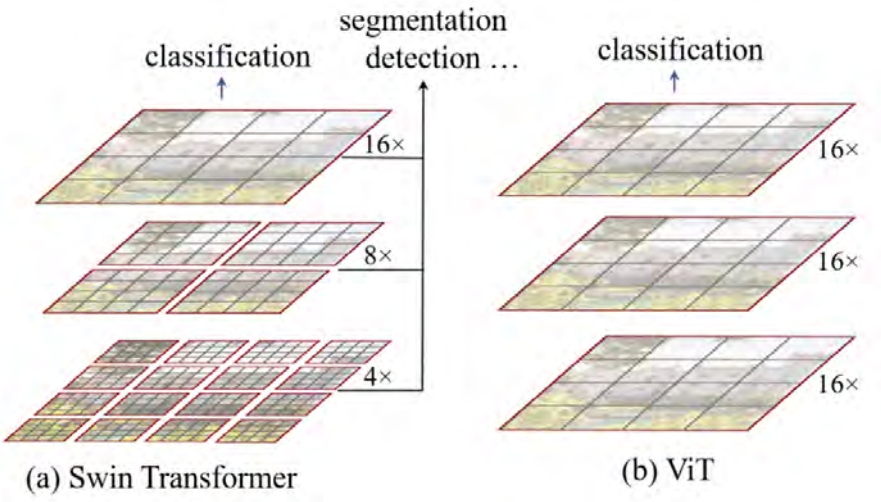


Figure 2.5: Comparison between Swin Transformer and Vision Transformer, where ViT creates feature maps of a single low resolution and quadratic computational complexity due to global computation of the self-attention mechanism, Swin Transformer has hierarchical feature maps with different resolutions and linear computational complexity due to computation of self-attention mechanism done in each local window [Liu+21c].

Swin Transformer has also been used in SR tasks such as "Image Restoration Using Swin Transformer (SwinIR)", with less amount of parameters, in their benchmarks, SwinIR still achieved better peak signal-to-noise ratio (PSNR) values comparing to EDSR [Lia+21]. In Figure 2.7, the SwinIR architecture is shown.

Hybrid Attention Transformer

In "Activating More Pixels in Image Super-Resolution Transformer" [Che+22b], Hybrid Attention Transformer (HAT) model is proposed to combine channel attention and self-attention schemes. In addition to that, they introduced an overlapping cross-attention module to enhance the interaction between neighboring window features to aggregate the cross-window information better, as shown in Figure 2.8. In Figure 2.9, the HAT

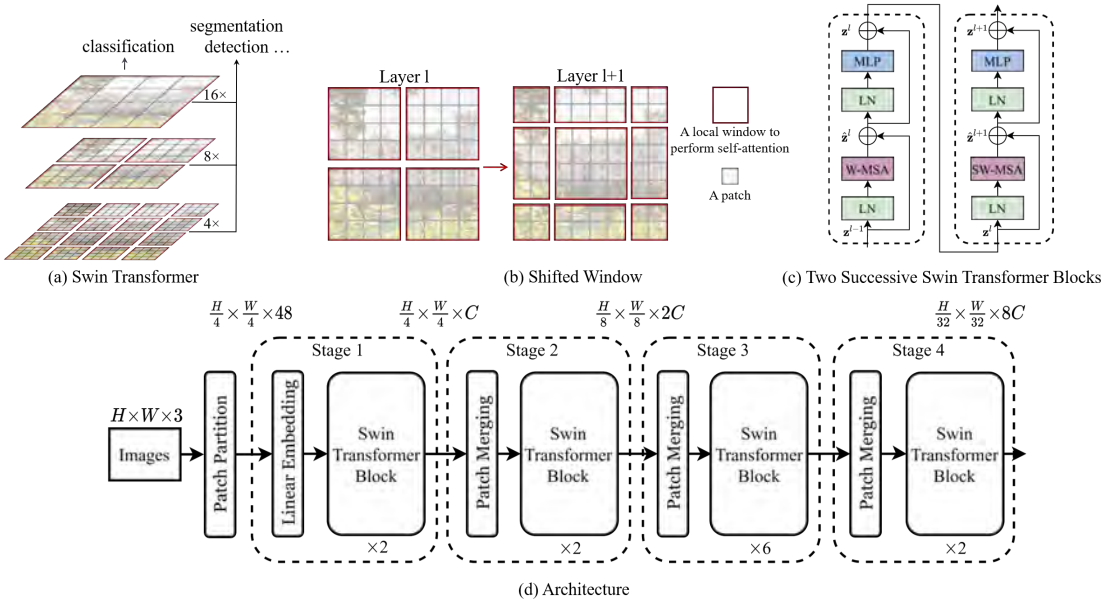


Figure 2.6: Architecture of Swin Transformer, where it uses shifted windows as multi-head self-attention in the blocks [Liu+21c].

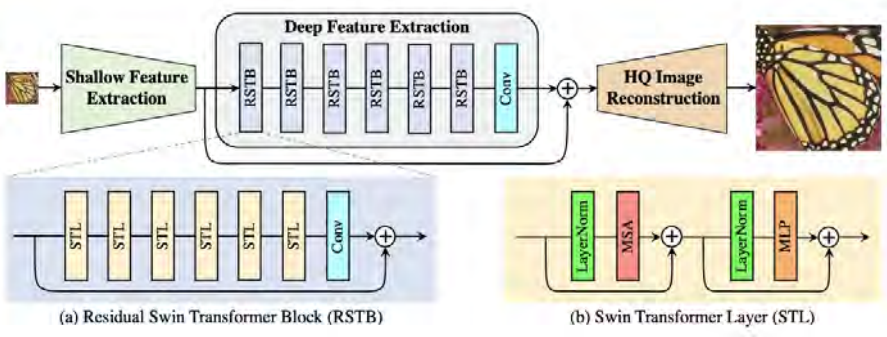


Figure 2.7: Model architecture of SwinIR, where it consists of shallow feature extraction, deep feature extraction layer with Residual Swin Transformer blocks with skip connections and finally an HQ Image Reconstruction layer to reconstruct SR image. [Lia+21]

architecture is shown. Also, being another transformer-based model such as Swin, their benchmarks show that HAT outperforms Swin, EDSR, and SRGAN architectures with being one of the state-of-the-art models for natural image SR [Che+22b]. However, the HAT model is not blind SR model, and it still lacks PSF degradation model to be able to reconstruct various SR images from real scene ultrasound images with different amounts of blurring.

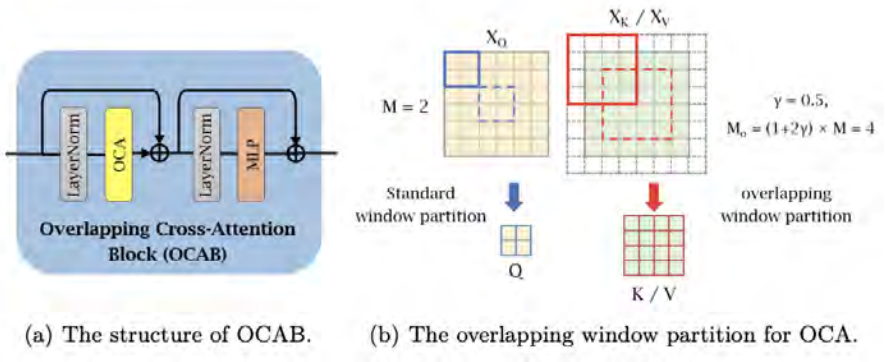


Figure 2.8: Overlapping Cross-Attention Block, similar to Swin Transformer block, but different in a way that generates key/value from a larger cross window than query since it's calculated with the overlapping window partition. [Che+22b]

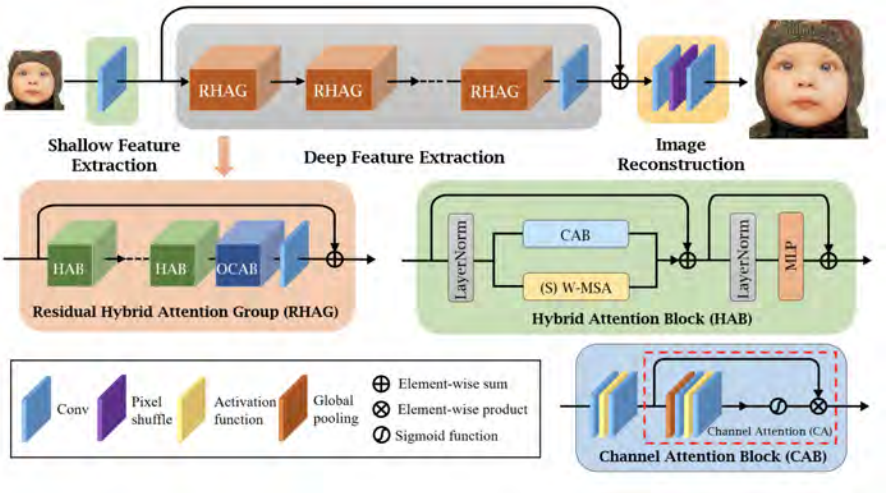


Figure 2.9: Model architecture of HAT [Che+22b]

2.1.3 Degradation-Aware Super-Resolution Methods

With the assumption of the real scene ultrasound images, authors of Progressive Residual Learning with Memory Upgrade for Ultrasound Image Blind Super-Resolution (PRLMU) state that the degradation blur process is more complex than bicubic sampling, therefore it needs to be solved by knowing the fitting degradation model [Liu+22]. Degradation-aware super-resolution methods such as blind SR is a more challenging task when compared to SR, where a low-resolution image is obtained without knowledge about the degradation process. Blind super-resolution algorithms first try to estimate the unknown blur kernel to use that information to reconstruct a HR image.

Super-Resolution Network for Multiple Degradations

Since most of the CNN based SR models assume that the degradation process is a simple bicubic downsampling, when actual degradation does not follow that assumption, such as in the case of ultrasound images, the SR model fails to reconstruct the HR image. In "Super-Resolution Network for Multiple Degradations" [ZZZ17], a SR model that can handle multiple degradation models such as downsampling, blurring, and noise is proposed with a dimensional stretching strategy. Their aim is to learn a single model to effectively handle multiple and spatially variant degradations while using synthetic data to train a model with high practicability [ZZZ17]. To solve the learning of a single model that can solve multiple degradations, the authors of Super-Resolution Network for Multiple Degradations (SRMD) propose a dimensional stretching strategy, where they used a single model to handle multiple degradations by stretching the input image to a higher dimension. In addition to that, they also managed to utilize synthetic data for the super-resolution problem, and they claimed that by choosing a better fitting degradation model than bicubic downscaling, learned SR model can return perceptually convincing results on actual LR images [ZZZ17].

For SRMD, the degradation model is represented as:

$$\mathbf{y} = (\mathbf{x} \downarrow_s) \bigotimes \mathbf{k} + \mathbf{n} \tag{2.2}$$

Their degradation model assumes an anisotropic Gaussian blur kernel, zero noise, and bicubic downsampler since dealing with blur and noise at the same time is a challenging task where it creates a large degradation space such as shown in Figure 2.10 and there exist not enough previous works about solving the task [ZZZ17].

So to solve the following problem, the authors of SRMD [ZZZ17] propose maximum-a-posteriori (MAP) estimation of the degradation model parameters, such that:

$$\hat{\mathbf{x}} = \arg\min_x = \frac{1}{2\sigma^2} ||(\mathbf{x} \bigotimes \mathbf{k}) \downarrow_s - \mathbf{y}||^2 + \lambda \Phi(\mathbf{x}) \tag{2.3}$$

where $\hat{\mathbf{x}}$ is the function for LR image $\mathbf{y}$, blur kernel $\mathbf{k}$, noise level σ, trade-off parameter λ and regularization parameter $\Phi\mathbf{x}$. Where the non-blind MAP solution to this problem

can be formulated into the following:

$$\hat{\mathbf{x}} = \mathcal{F}(\mathbf{y}, \mathbf{k}, \sigma, \lambda; \Theta) \tag{2.4}$$

where Θ is the parameters of the MAP. Then the λ can be absorbed into σ and final goal will be:

$$\hat{\mathbf{x}} = \mathcal{F}(\mathbf{y}, \mathbf{k}, \sigma; \Theta) \tag{2.5}$$

, which is still more complex than $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{y}; \Theta)$ to learnable via CNN since the three inputs $\mathbf{y}, \mathbf{k}, \sigma$ have different dimensions. Therefore SRMD [ZZZ17] propose the dimensionality stretching strategy, where it vectorizes blur kernel to a $p^2 * 1$ space and then projected onto a t dimensional space by PCA, concatenated by the noise level and stretched to degradation maps of $W * H * (t + 1)$.

After creating the degradation maps, the SRMD [ZZZ17] model takes the LR image and the degradation maps as input such that $W * H * (C + t + 1)$. Then convolution blocks, batch normalization, and ReLU activation are applied to perform the nonlinear mapping. Finally, a last convolution converts the image size of $W * H * s^2C$ to a single image in the size of $sW * sH * C$ where s is the scaler from LR to HR [ZZZ17]. The convolutional layers number is 12, and feature maps in each layer are set to 128 [ZZZ17]. An overview of the SRMD architecture can be shown in Figure 2.11

Finally, authors [ZZZ17] claim that learning a blind SR model when blur kernel is complex, such as when motion blur is in the LR image instead of Gaussian blur, does not perform well and explains this phenomenon by pixel-wise averaging [Led+16] problem where shifting the image to the left by one pixel and shifting blur kernel to the right by one pixel yield the LR image and argues that blind SR models cannot generalize easily to unseen degradations.

Zero Shot Super-Resolution

These methods named Zero-Shot Super-Resolution using Deep Internal Learning (ZSSR) [SCI17] and Perception Consistency Ultrasound Images Super-Resolution via Self-Supervised CycleGAN (USSSCSR) [Liu+21a] work with "zero-shot" principle shown in Figure 2.12 where a model learns from the internal recurrence of information from a single image by training an image-specific CNN.

ZSSR creates an image-specific CNN by doing augmentations on that single image which are named "HR fathers" and then downscaled by scale factor s to obtain "LR sons". Then this group of data is transformed by 4 rotations and vertical/horizontal rotations, which creates 8 times more data, and finally, scaling is done gradually (for example, for 8x scaling, scaling may be done by 2x scaling for 3 times, etc.) if scale factors are large, to increase robustness.

ZSSR employs a fully convolutional network of 8 hidden layers consisting of 64 channels with ReLU activations with residuals between interpolated LR son and HR

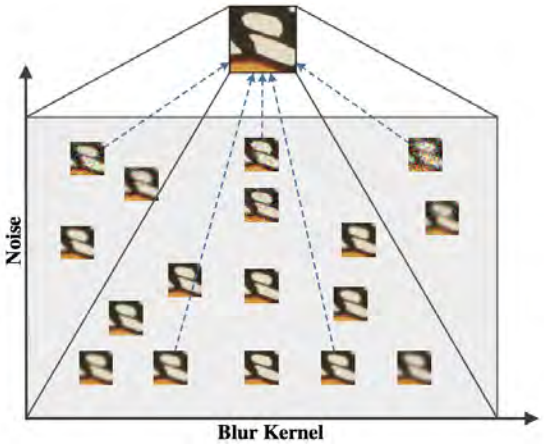


Figure 2.10: Different degradation levels of noise and blur kernel creates a lot of LR images where most of the SR algorithms seek one LR to one HR mapping due to the bicubic downsampling, which creates complexity. [ZZZ17]

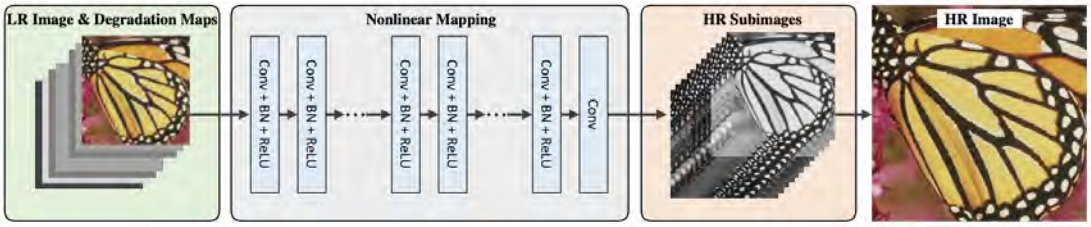


Figure 2.11: Model architecture of SRMD, where it takes image and degradation maps feed SRMD model consisting of blocks of "Conv+BN+ReLU" then creates Subimages to a single HR image [ZZZ17]

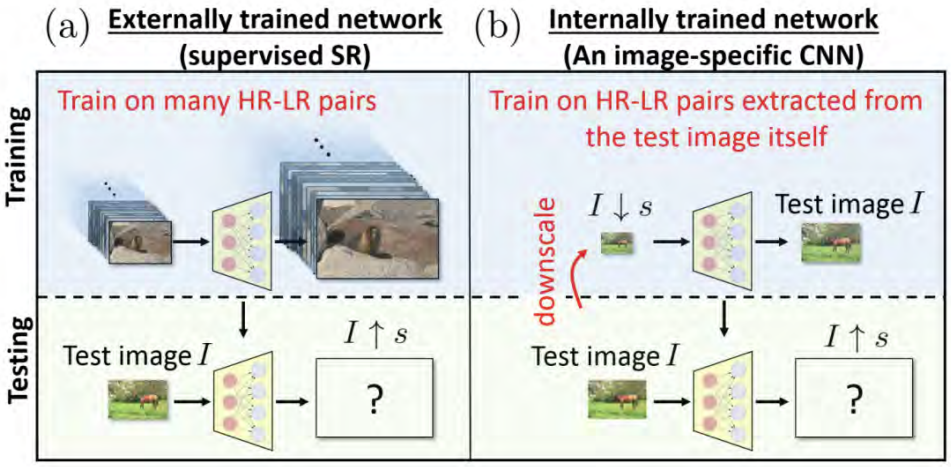


Figure 2.12: Zero-Shot is a principle based on having a single image where examples are extracted internally from that image to learn how to synthesize SR image from its coarser resolutions (patches) to create itself as SR image. Where traditional SR networks learn by many pairs of LR and HR images. [SCI17]

father. It uses L_1 loss with Adam [KB14] optimizer with a learning rate of 0.001 and stops at a learning rate of $10^{-}6$ by dividing the learning rate by 10 where the standard deviation of the linear fit of the reconstruction error is a factor greater than the slope of the fit. Overview of ZSSR [SCI17] can be seen in Figure 2.13

Iterative Kernel Correction

Iterative Kernel Correction for Ultrasound Image Super-Resolution (IKC) [Gu+19] is a blind SR method for blur kernel estimation, which tries to predict and correct the blur kernel iteratively. Authors argue that taking the concatenation of the image and blur kernel (degradation maps) as input is not an optimal solution. Therefore, they propose a blind super-resolution method that tries to solve the following:

$$\theta_p = \arg\min_{\theta_p} ||k - \mathcal{P}(I^{LR}; \theta_p)||_2^2 \tag{2.6}$$

where k is the blur kernel and $\mathcal{P}(I^{LR})$ is the predictor which estimates blur kernel k'. Since the accurate estimation of k is impossible, it may lead to kernel mismatch, which can cause artifacts such as shown in Figure 2.14. Therefore, to correct the kernel mismatch, authors [Gu+19] propose a corrector structure:

$$\theta_c = \arg\min_{\theta_c} ||k - (\mathcal{C}(I^{SR}; \theta_c) + k')||_2^2 \tag{2.7}$$

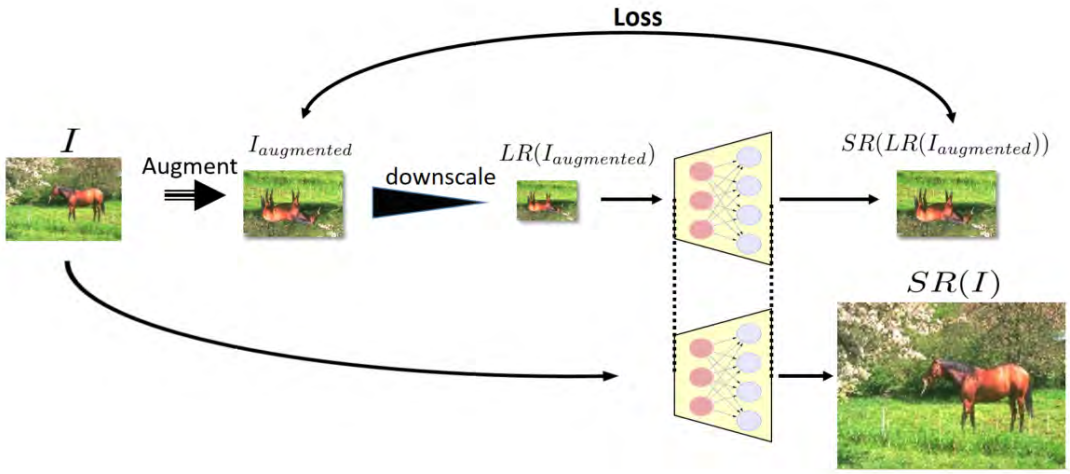


Figure 2.13: ZSSR creates "HR fathers" as augmenting images, then downsamples it to create "LR sons" to compute the SR information in patches of a single image to use that information to upscale the image given to real super-resolution [SCI17].

Since the corrector may overshoot if done at once and may lead to another kernel mismatch, the structure is used to correct the kernel gradually, so iteratively. Therefore the overall architecture looks as in Figure 2.15, also the overview of the $\mathcal{P}$ and $\mathcal{C}$ networks are shown in Figure 2.16

2.2 Ultrasound Image Super-Resolution

In this section, some of the recent works on SR, especially for ultrasound images, are reviewed.

2.2.1 Deep Convolutional Neural Network for Ultrasound Super-Resolution

DECUSR is one of the earliest SR methods developed specifically for B-mode ultrasound images. Similar to the EDSR and SRGAN, the architecture is based on the CNN layers. The DECUSR architecture is shown in Figure 2.17. The architecture consists of a feature extraction module, repeating blocks, and a concatenation/upsampling layer to reconstruct the image. According to their benchmarks, DECUSR outperforms EDSR and SRCNN architectures regarding ultrasound images. While this model is remarkable in terms of learning-based ultrasound SR, when compared to Blind SR models, their experiment lacks PSF degradation. They used a bicubic downscaled dataset to reconstruct the SR image. As the resolution degradation blur process of an actual scene is intricate and unknown, according to Liu [Liu+22], this assumption is

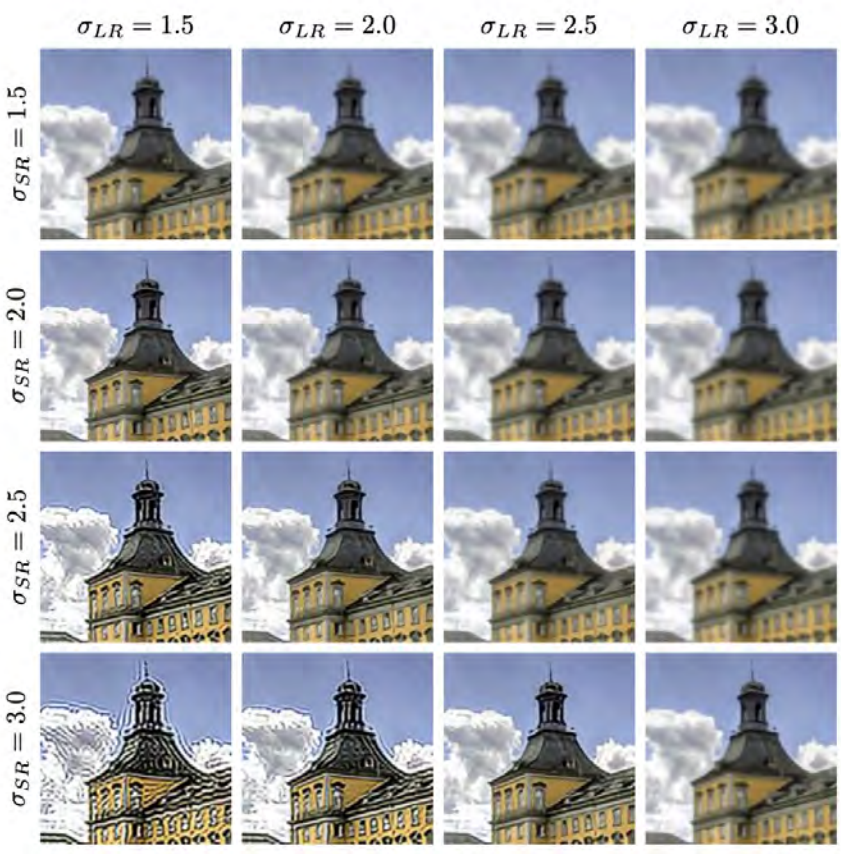


Figure 2.14: Blur kernel mismatch where σ_{LR} denotes the downsampling kernel and σSR denotes the kernel used to super-resolve. When $\sigma LR > \sigma SR$, it creates still blurry images, and when $\sigma LR > \sigma SR$, it creates over-sharpened images. [Gu+19]

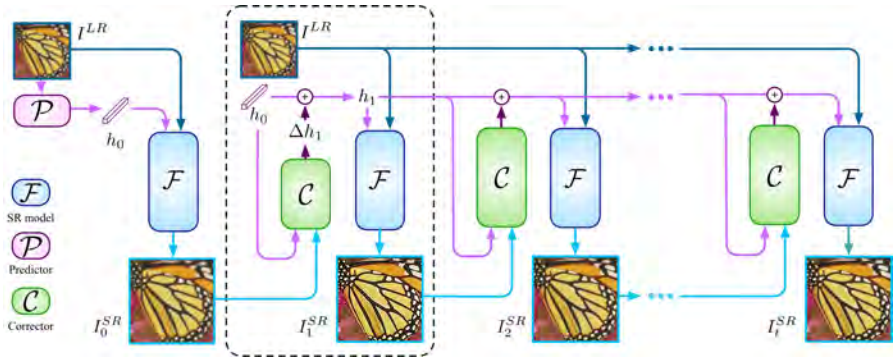


Figure 2.15: Predictor predicts the blur kernel which is used by SR model to generate SR image, which fed again into corrector with the given blur kernel to calculate the mismatch of the blur kernel and new blur kernel fed again iteratively until the proper kernel is found [Gu+19].

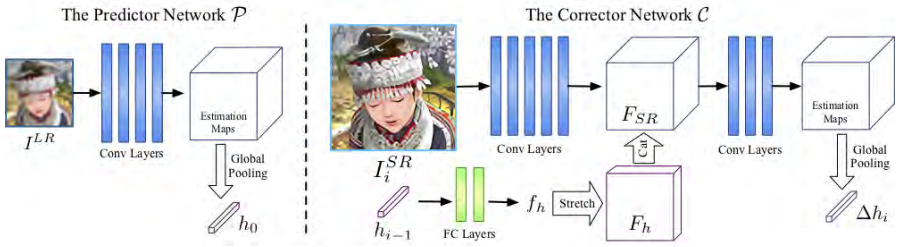


Figure 2.16: Overview of the predictor and corrector modules used in IKC [Gu+19].

not valid for ultrasound imaging.

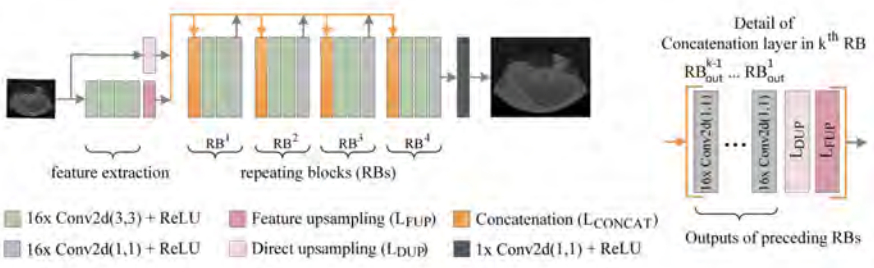


Figure 2.17: Model architecture of DECUSR [TB20]

2.2.2 Perception Consistency Ultrasound Image Super-Resolution via Self-Supervised CycleGAN

In addition to ZSSR [SCI17], USSSCSR [Liu+21a] proposes an ultrasound image super-resolution network combined CycleGAN [Zhu+17] into ZSSR [SCI17], which introduces cycle consistency, improves the image quality. Similar to ZSSR [SCI17], this model also creates "HR fathers" by augmenting the original image and creates "LR sons" by down-sampling them. Then the network is replaced by CycleGAN [Zhu+17], which utilizes a multi-scale structure for the generator part and considers perception consistency by training LR to HR, then using that HR to LR back. After the training of the CycleGAN is completed, the actual image is sent as LR input to obtain SR reconstruction. By utilizing the multi-scale structure in the LR to SR generator, such as shown in Figure 2.18, they are simulating the sensation of low-frequency ultrasound images by downscaling the images for the same part, since some of the details are lost when downscaled, similar to the low-frequency ultrasound which aims to reconstruct the image in different scales.

In addition to LR - SR generator, for the cycle consistency, to eliminate the artificial and redundant details introduced in image generation and fulfill the CycleGAN-styled pipeline, they also designed a HR - LR image generation network by feeding the high-quality image and accompanying Gaussian noise as input as shown in Figure 2.19.

For the discriminator part, they employed a PatchGAN style discriminator to calculate adversarial loss, such shown in Figure 2.20.

The total loss function of USSSCSR [Liu+21a] as follows:

$$\mathcal{L}_{\text{total}} = \alpha\mathcal{L}_{\text{pixel}} + \beta\mathcal{L}_{\text{perceptual}} + \gamma\mathcal{L}_{\text{adversarial}} + \eta\mathcal{L}_{\text{cycle}} \tag{2.8}$$

where $\mathcal{L}_{\text{pixel}}$ is the pixel-wise loss, $\mathcal{L}_{\text{perceptual}}$ is the perceptual loss coming from VGG [SZ14], $\mathcal{L}_{\text{adversarial}}$ is the adversarial loss, $\mathcal{L}_{\text{cycle}}$ is the cycle consistency loss. α, β, γ, η are the weights of the loss functions.

As an overview, Figure 2.21 shows the whole pipeline of USSSCSR [Liu+21a].

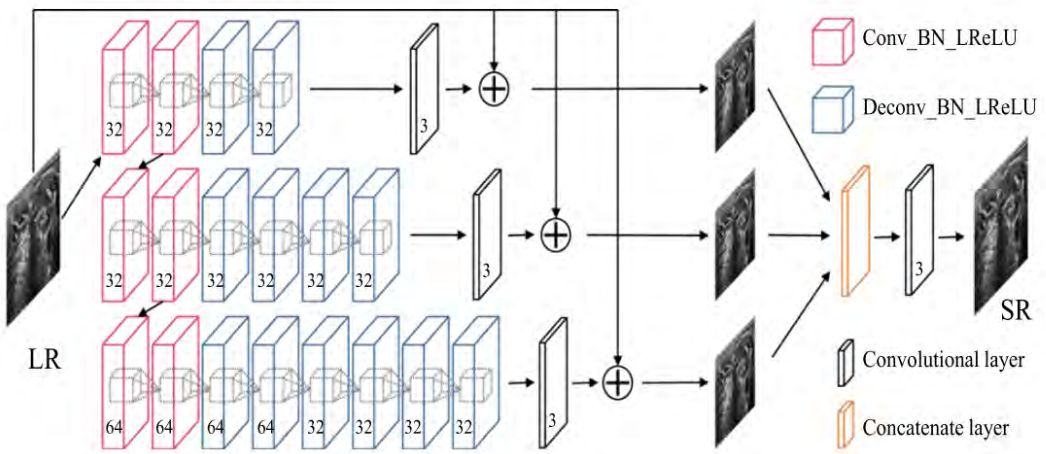


Figure 2.18: Multi-scale generator employs three different scales to recover the SR image [Liu+21a].

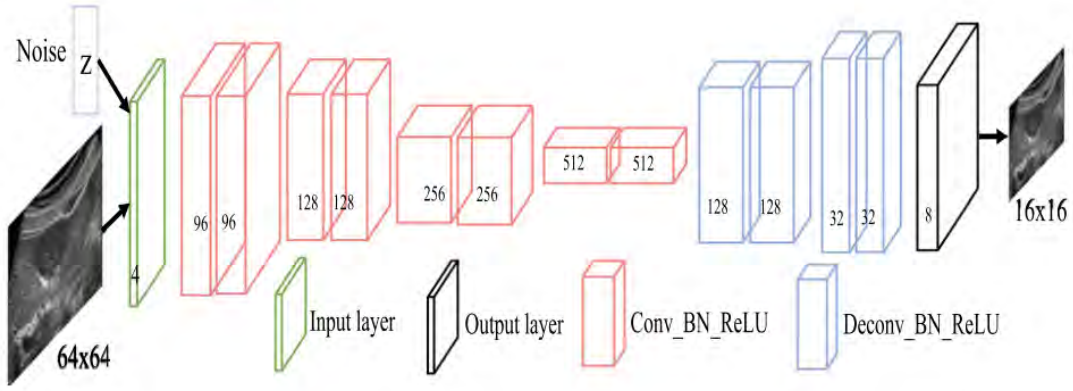


Figure 2.19: HR to LR Generator of USSSCSR, where it takes HR image and Gaussian noise to output LR image. [Liu+21a].

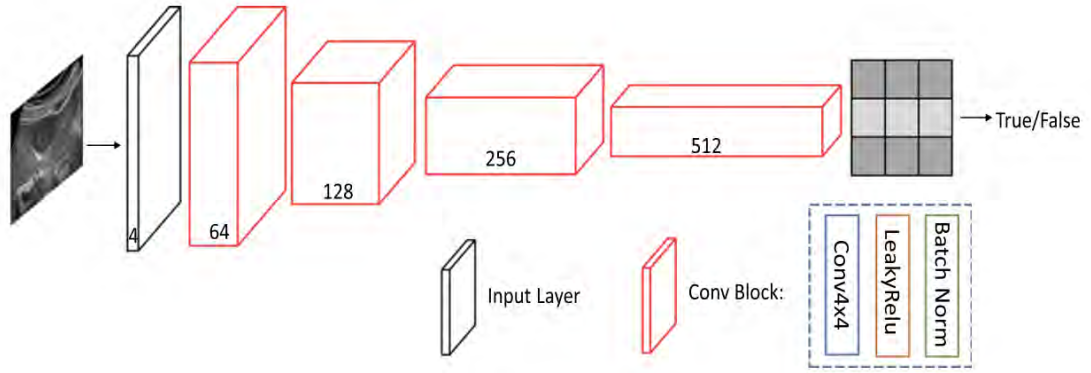


Figure 2.20: Structure of the discriminator of USSSCSR [Liu+21a].

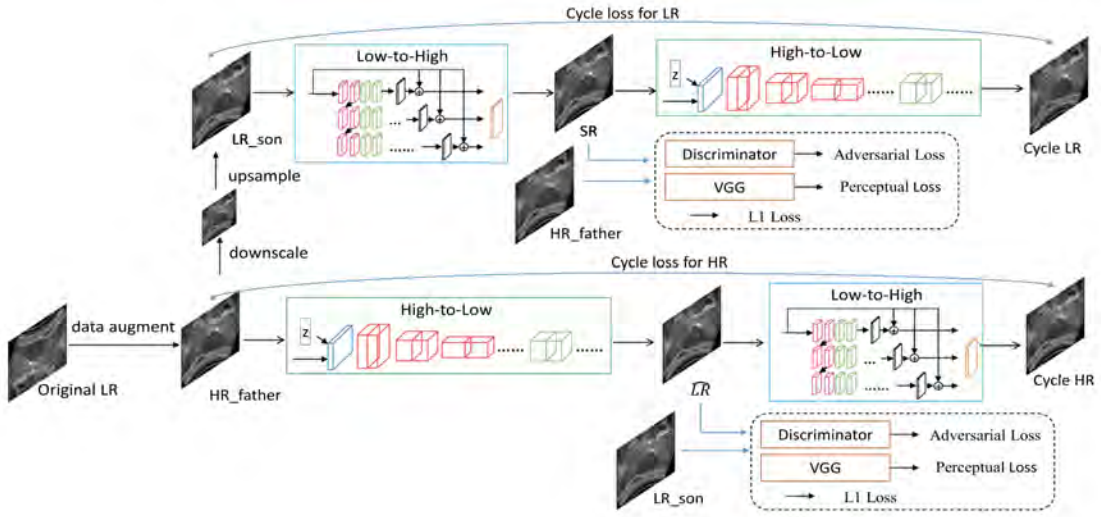


Figure 2.21: USSSCSR pipeline [Liu+21a].

2.2.3 Progressive Residual Learning with Memory Upgrade for Ultrasound Image Blind Super-Resolution

PRLMU [Liu+22] is a SR method which is used to enhance the quality of the ultrasound images. Similar to IKC [Gu+19], this model also assumes the existence of Gaussian blur in the images of ultrasound without additive noise to reduce the complexity, then instead of using a corrector, they [Liu+22] try to solve the following problem:

$$r = y - (x * k) \downarrow_s \tag{2.9}$$

where r is the residual, y is the degraded image, x is the SR image, k is the blur kernel, $\downarrow_s$ is the downsampling operator. The goal is to find the residual r, the difference between the degraded image and the SR image. Therefore, the authors introduce residual learning to transform the SR problem to the progressive restoration of residuals iteratively, such shown in Figure 2.22. In addition to that, PRLMU seeks for highest spatial attention score to determine the blur kernel by calculating the spatial attention map such as shown in Figure 2.23 and also proposes Improved Channel Attention Block (ICAB) to upsample LR image to create SR image as in Figure 2.24. Finally, by using memory upgrade PRLMU [Liu+22] aims to store and update residuals to obtain finer details.

To summarize, the PRLMU [Liu+22] architecture does the following operations:

$$\begin{aligned} k &= Estimate(I_{LR}) \\ M_i &= [M_{i-1}, upsampling(I_{r_{i-1}}, M_{i-1})] \\ I_{SR_{i-1}} &= sum(M_i) \\ I_{r_i} &= I_{LR} - (I_{SR_{i-1}} * k) \downarrow \end{aligned} \tag{2.10}$$

For the loss, it uses the following loss function:

$$\begin{aligned} \mathcal{L}_{total} &= \ell_{sr} + \ell_{lr} + \ell_{kernel} \\ \ell_{sr} &= ||I_{HR} - G(I_{LR})||_1 \\ \ell_{lr} &= \sum_{i=1}^{n} ||\alpha_i (I_{LR} - D_i(SR_i))||_1 \\ \ell_{kernel} &= \sum_{c=1}^{m} (1 - y_{I_{LR},c} \log(p^v_{I_{LR},c})) + \sum_{c=1}^{n} (1 - z_{I_{LR},c} \log(p^s_{I_{LR},c})) \end{aligned} \tag{2.11}$$

where ℓ_{sr} is the ℓ_1 pixel-wise loss by the HR image and SR image generated. ℓ_{lr} is the residual learning loss, where it tries to optimize residue, D_i is the residual learning module for ith stage, and α_i is the weight coefficient set to 0.1 for each stage, this process is exampled in Figure 2.22.ℓ_{kernel} is the kernel learning loss, which is calculated as softmax cross entropy, where m, n are respectively the number of categories for

the variance and kernel size and $y_{I_{LR},c}, z_{I_{LR},c}$ are the one-hot encoding values for the variance and kernel size, $p^v_{I_{LR},c}, p^s_{I_{LR},c}$ are the probability of the kernel variance and size estimated from the LR image.

Finally, the whole architecture can be seen in Figure 2.25.

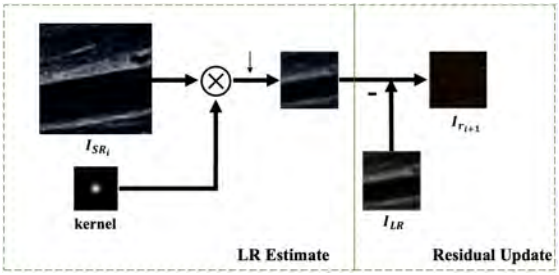


Figure 2.22: Residual learning mechanism tries to update residue gradually. [Liu+22].

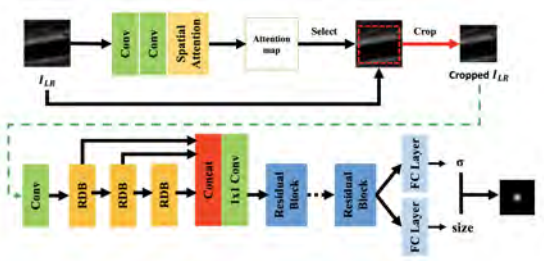


Figure 2.23: Spatial attention mechanism tries to find the highest spatial attention score to determine the blur kernel. [Liu+22].

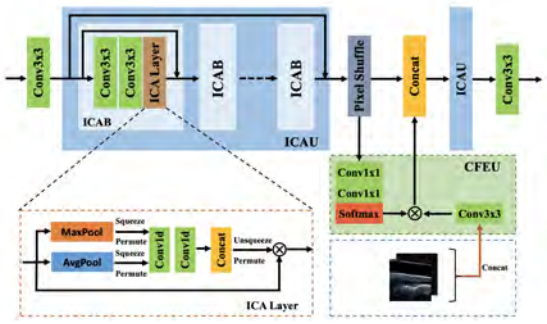


Figure 2.24: ICAB module is used for upsampling the LR image to create the SR image. [Liu+22].

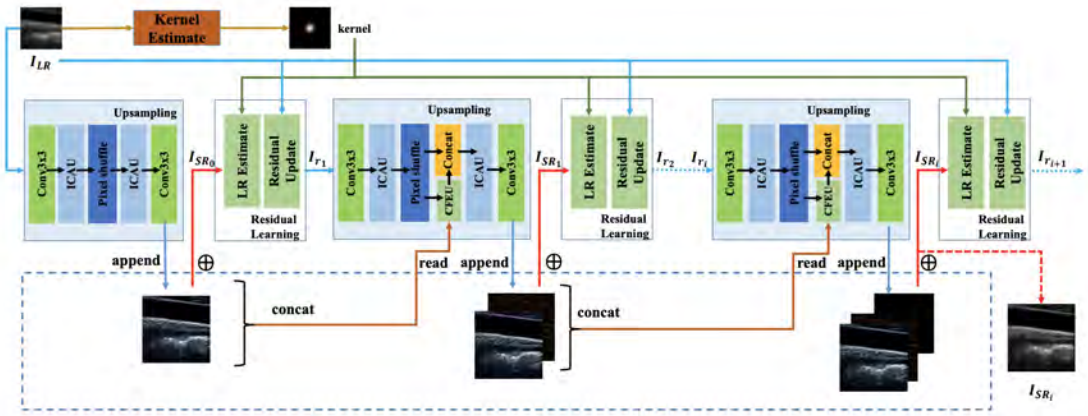


Figure 2.25: Overview of the PRLMU [Liu+22] architecture.

3 Dataset

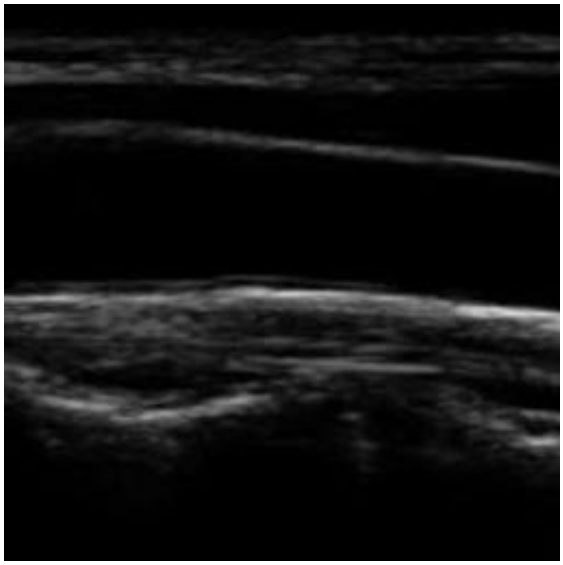
Since one of the older approaches, PRLMU, does not have a publicly available training code to have a common benchmark between these methods, in this thesis, common cartoid artery ultrasound (CCA-US) dataset is used as one of the datasets. The CCA-US data contains 84 B-mode images acquired by Sonix OP ultrasound scanner [Zuk+13]. Some images with different capturing settings exist, such as shown in Figure 3.1. Since it may create a bias in training, the images similar to the test set have been deleted from the training set. In addition to that, some images are smaller than $256 * 256$, and these are deleted since we are randomly cropping images to $256 * 256$ as HR images. In total, 79 ultrasound images with width and height bigger than at least $256 * 256$. Randomly selected 61 images are used as the training set, 9 of the images are used as the validation set, and 9 of the images are used as the test set. An overview of the images in the CCA-US dataset [Zuk+13] can be seen in Figure 3.2

In addition, as a second dataset, Breast Ultrasound Images Dataset (BUSI) is used. BUSI is another public dataset collected in 2018, consisting of 780 images with an average image size of $500 * 500$ pixels having breast ultrasound images among women aged between 25 and 75 [Al-+20]. Randomly 120 images are taken from with a resolution bigger than $256 * 256$, randomly cropped $256 * 256$ pixels in each image and 100 of them are used as the training set, 10 of them are used as the validation set, and 10 of them are used as the test set. An overview of the images in BUSI dataset [Al-+20] can be seen in Figure 3.3

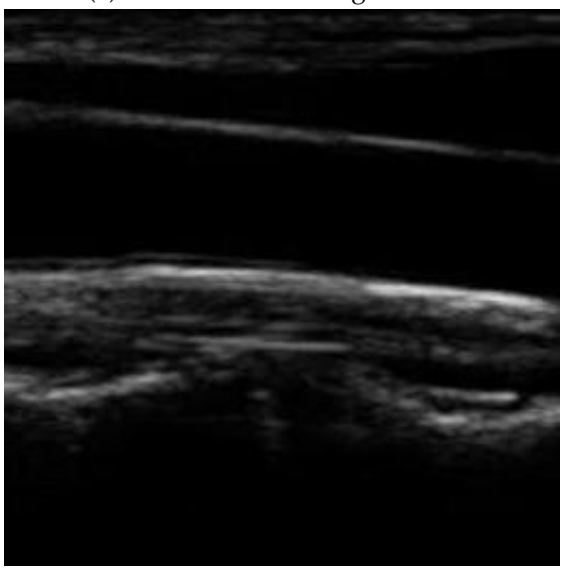
3.1 Data Preprocessing

By following the same claims of SRMD and PRLMU [ZZZ17; Liu+22], the LR images are created synthetically by downscaling the HR images with bicubic interpolation and synthetically adding blur and omitting noise. In that way, we aim to train a robust network that can learn different levels of blur and check if creating synthetic degradation helps the SR network. Therefore, we will categorize the datasets in two different levels for the experiments: "no-blur (bicubic degraded) images" and "Gaussian blurred images".

For preprocessing, by taking one of the images from these datasets [Zuk+13; Al-+20], it is possible to see that the images are not the same size. To make the images at the same size, these images are randomly cropped to the same size to the size of $256 * 256$ for training and validation sets. Then the LR images are created by downscaling the



(a) An ultrasound image in CCA



(b) Another image, probe moved to the right

Figure 3.1: Two different ultrasound images in the CCA dataset taken in the similar case [Zuk+13]. This situation might create a bias if one fed to the test set and another to the training set.

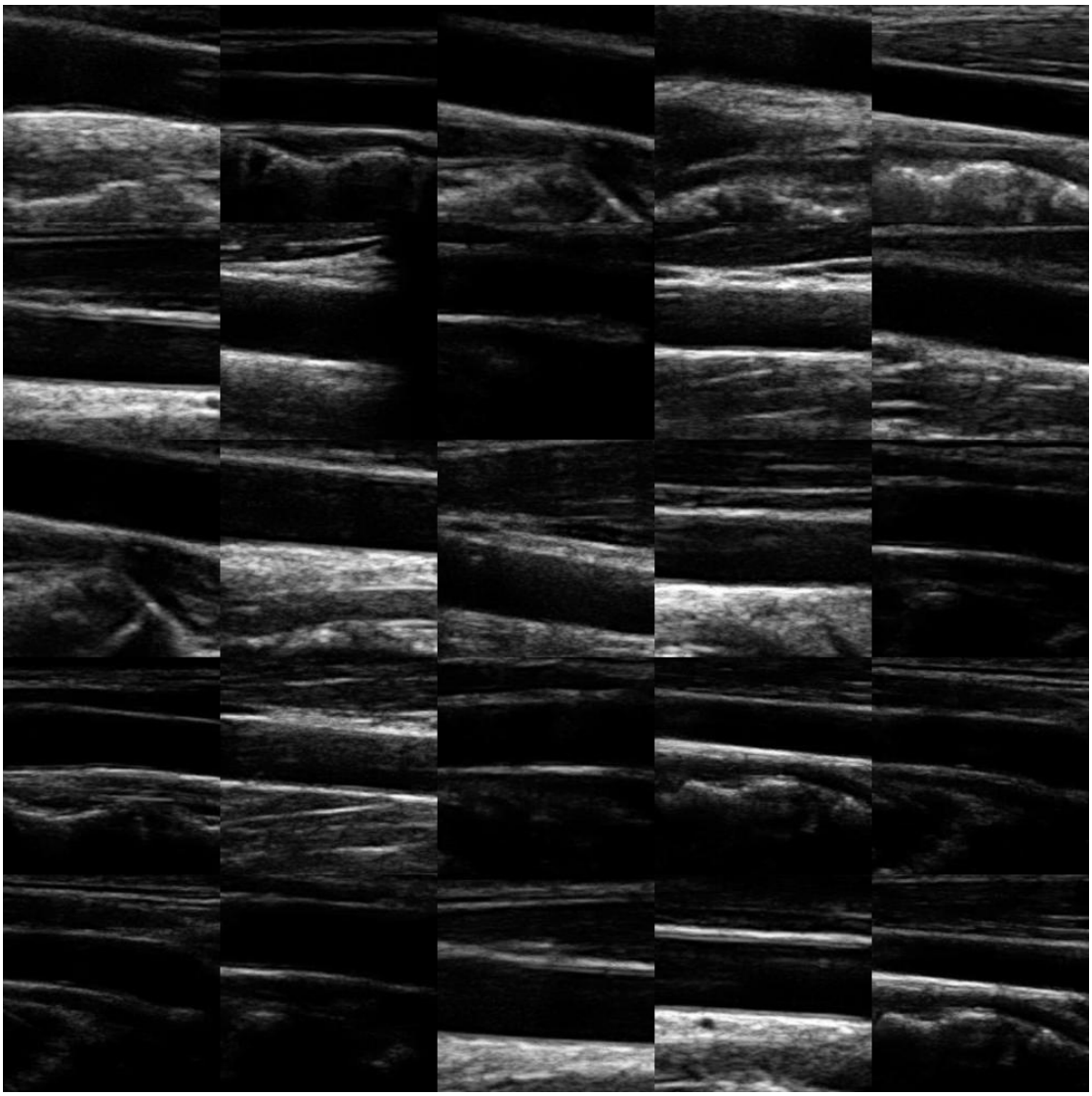


Figure 3.2: An overview of the images in CCA dataset [Zuk+13]

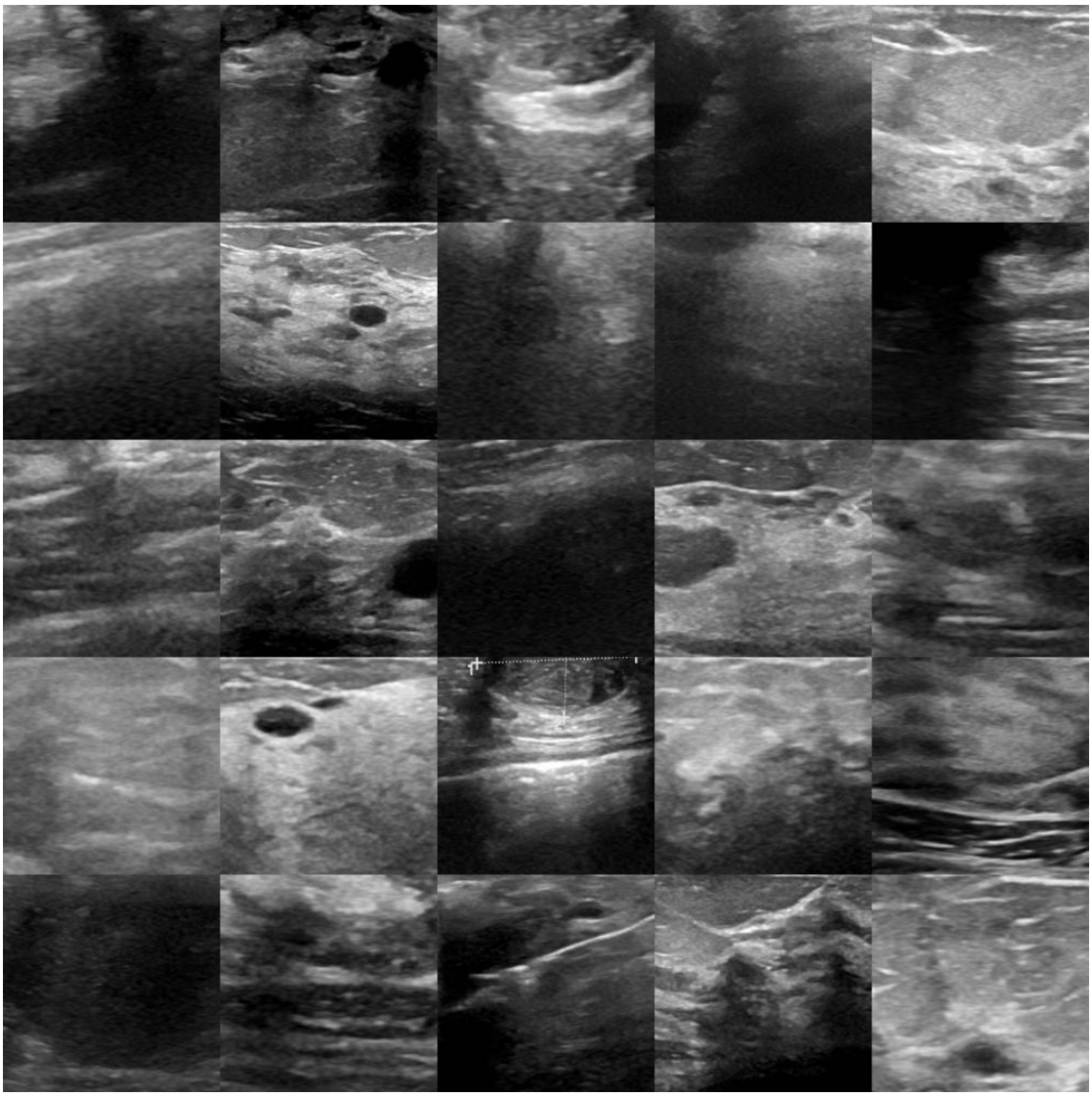


Figure 3.3: An overview of the images in BUSI dataset [Al-+20]

HR images with bicubic interpolation. The LR images are downsampled to the size of $64 * 64$ to create the no-blur category. Then a Gaussian blur with a kernel size of 21 pixels and a sigma between 1.8 to 3.2 pixels is applied in 8 different levels to create the second category, blurred images. Then all of these LR images are bicubically upscaled to $4x$ back to the exact resolution for fake-SR copies. For the BUSI dataset, in the validation set, 10 different pictures with sigma 2.0 and 3.0 totaled 20 images. Also, in the test set, 10 pictures with sigma 2.0 and 3.0 totaled 20 images. Overall, there exist 2 different categories on 2 different datasets. For the CCA dataset, 9 different pictures with sigma 2.0 and 3.0 in the validation set totaled 18 images. Also, 9 pictures with sigma 2.0 and 3.0 totaled 18 images in the test set. By having a small but enough amount of testing and validation images, we can better understand the network's performance for the different amounts of Gaussian blurring. Overall, there exist 2 different categories (i.e., bicubic downsampled and Gaussian blurred) on 2 different datasets (i.e., CCA and BUSI). The whole process of generating datasets can be seen in Figure 3.4.

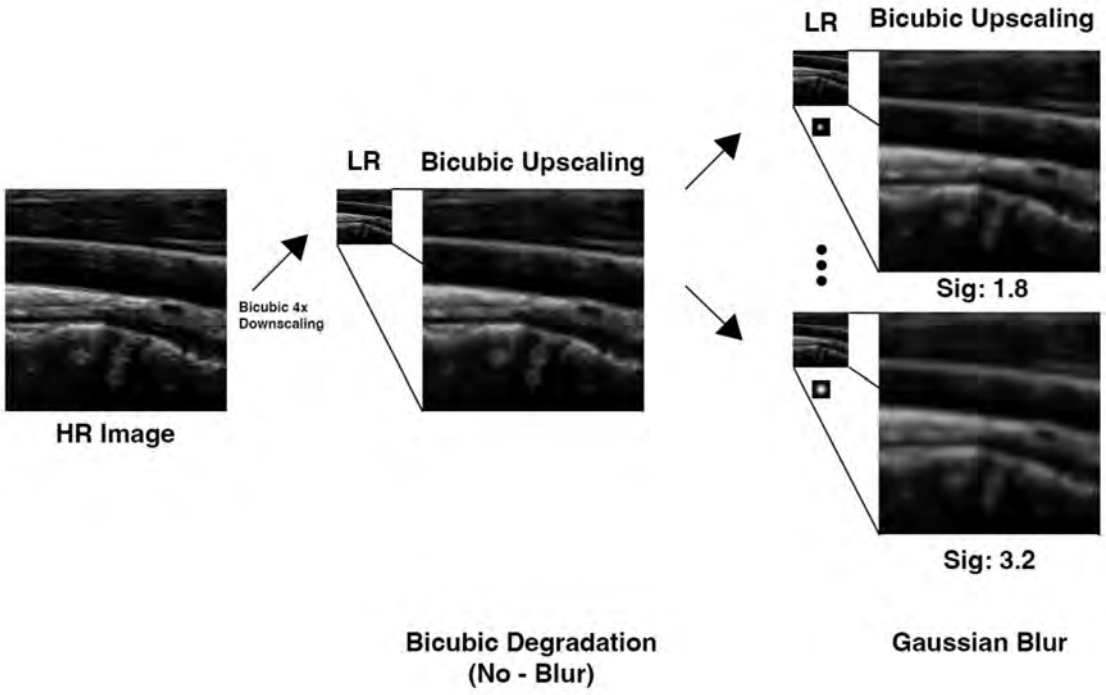


Figure 3.4: Dataset preprocessing, synthetically creating two categories by bicubically downgrading and applying various amounts of Gaussian blur.

4 Method Overview

In this thesis, we aim to enhance the resolution of PSF degraded (blurry) ultrasound images using super-resolution (SR) techniques. We explored two approaches for SR: SR methods and degradation-aware SR methods. While SR methods rely on a known degradation model, such as bicubic downsampling, degradation-aware SR methods do not assume any specific degradation model.

After reviewing the different methods for ultrasound super-resolution, since the classic SR methods lack the degradation awareness, and blind SR models might not realize unseen degradation models, such authors of SRMD [ZZZ17] argued, we propose to use a deblurring network, such as DeblurGANv2 [Kup+19] or NAFNet [Che+22a] to realize blur and noise weights to fine-tune the SR image without having to deal with precomputed blur kernels as blind SR models do.

The pipeline is to restore the LR image to a fake SR image using bicubic upscaling. The restored image is then fed to the deblurring network to generate the final high-resolution (HR) image.

Since the super-resolution problem is modeled as [Liu+22]:

$$y = (x * k) \downarrow_s + n \tag{4.1}$$

where n is just the noise, k is the blur/degradation kernel, and s is the scale factor for the (bicubic) downsampling. The goal is to find the HR image y from the LR image x. Therefore the difference between the LR image y and the original HR image x should be minimized in terms of PSNR and structural similarity (SSIM) which will be explained in the Section 6.2.

This approach offers the advantage of utilizing both the information from the LR image and the prior knowledge of the various amounts of blur kernels to enhance the resolution of the image flexibly. By using a deblurring network, we can effectively reduce the blur, noise, and other artifacts in the image, resulting in a high-quality HR image.

In addition to that approach, we will also be exploring the use of the recent natural image SR models such as HAT [Che+22b] and SwinIR [Lia+21] and our other proposal, which is HAT-NAF mixture that aims to simplify HAT [Che+22b] network with nonlinear activation-free [Che+22a] blocks to see if they can be used to enhance the resolution of ultrasound images such shown in Figure 4.1. We will be comparing the performance of the two approaches and see which one performs better.

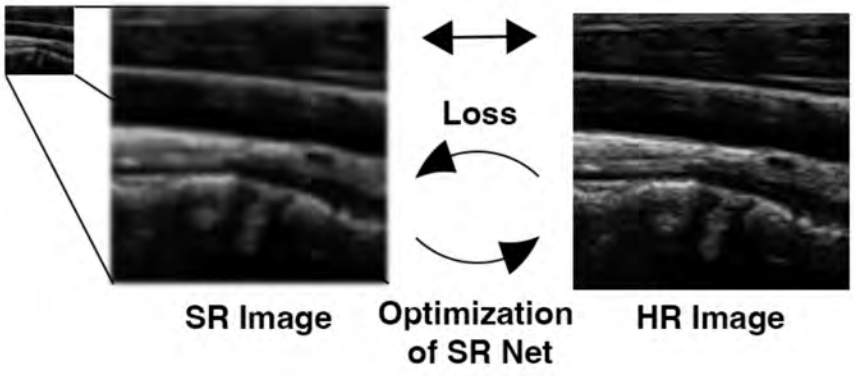


Figure 4.1: Model Overview of Super-Resolution Networks to fine-tune degraded image as super-resolved image.

5 Network Architecture

Since our approach is to optimize the difference between a bicubically upscaled fake super-resolution image and HR ground truth image to realize the deblur and denoising parameters to enhance the resolution of the PSF degraded (blurry) ultrasound images, we need to first introduce some of the deblurring networks. We will then introduce the two deblurring networks we used in this thesis: DeblurGAN [Kup+18] and DeblurGANv2 [Kup+19]. Finally, we will introduce the NAFNet [Che+22a], a new deblurring network we proposed in this thesis.

5.1 DeblurGAN

In this thesis, we focus on deep learning-based approaches for image deblurring and denoising to fine-tune a bicubically upscaled image to reconstruct a SR image as similar to the HR ground truth image, we introduce DeblurGAN as our first approach, which is replaced by DeblurGANv2 later on, but to comprehend the concept, DeblurGAN is essential.

DeblurGAN is a generative adversarial network (GAN) proposed by O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas in 2018 [Kup+18] to learn the mapping between blurred and sharp images. The model consists of a generator network that generates deblurred images and a discriminator network that distinguishes between real and generated images. By training the generator and discriminator networks in an adversarial manner, DeblurGAN learns to generate high-quality deblurred images. In that way, DeblurGAN benefits from using GANs, which are famous for preserving texture details in images, and creating perceptually convincing outputs when compared to the plain mean squared error (MSE) loss optimization shown in Figure 5.1.

DeblurGAN is designed as an end-to-end learning method for blind motion deblurring of a single photograph. DeblurGAN utilizes a Markovian (PatchGAN) discriminator which penalizes a structure at the scale of patches by basically classifying each $N * N$ area of an image as real or fake with a convolutional network shown in Figure 5.3 [Iso+16]. In Figure 5.2, generator architecture is shown [Kup+18].

Since DeblurGAN is a generative adversarial network, the generator tries to create an unblurred, high-quality image. The discriminator tries to prove if an image is real or fake, distinguishing between reconstructed SR image and HR image samples. Therefore optimization game between generator G and discriminator D is described as a minimax objective:

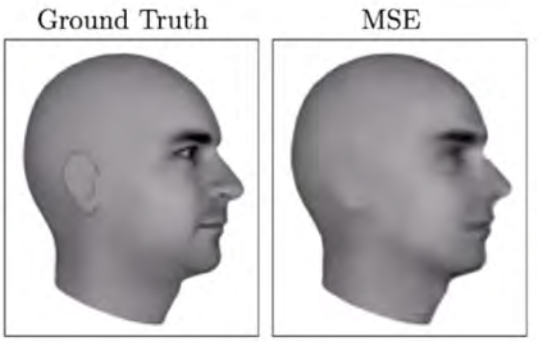


Figure 5.1: When MSE loss is used, the network will tend to predict averaged outputs, resulting in blurry and visually unpleasant images. [LKC16]

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[\log D(x)] + \mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[\log(1-D(\tilde{x}))] \tag{5.1}$$

where $\mathbb{P}_r$ is the distribution of the data and $\mathbb{P}_g$ is the distribution of of the model and $\tilde{x} = G(x), z \sim P(z)$. Solving this minimax objective as vanilla GAN function causes issues such as mode collapse, vanishing gradient, and many more, therefore in DeblurGAN's discriminator, WGAN loss based on Wasserstein-1 distance is used as a design choice [Kup+18]. The WGAN-GP loss function is defined as following [Gul+17]:

$$min_G, max_{D\in\mathcal{D}} \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)] - \mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})] + \lambda \mathbb{E}_{\tilde{x}\sim\mathbb{P}_{\tilde{x}}}[||\nabla_{\tilde{x}} D(\tilde{x})||_2 - 1]^2 \tag{5.2}$$

where $\mathcal{D}$ is the set of 1-Lipschitz functions and λ is the gradient penalty coefficient. The gradient penalty is used to enforce the Lipschitz constraint on the discriminator where the idea is to approximate $K \cdot W(P_r, P_\theta)$ where K is Lipschitz constant and W is the Wasserstein distance [Kup+18].

Also, for generator loss, a combination of content and adversarial loss is used:

$$\mathcal{L} = \mathcal{L}_{adversarial} + \lambda \mathcal{L}_{content}, \lambda = 100 \tag{5.3}$$

$$\mathcal{L}_{adversarial} = \sum_{n=1}^{N} -D_{\theta_D}(G_{\theta_G}(I^B)) \tag{5.4}$$

$$\mathcal{L}_{content} = \frac{1}{W_{i,j}, H_{i,j}} \sum_{i=1}^{W_{i,j}} \sum_{j=1}^{H_{i,j}} (\phi_{i,j}(I^S)_{x,y} - \phi_{i,j}(G_{\theta_G}(I^B))_{x,y})^2 \tag{5.5}$$

where I^B is the input blurry image, I^S is the ground truth sharp image, G_{θ_G} is the generator network, D_{θ_D} is the discriminator network, $\phi_{i,j}$ is the feature extractor network, $W_{i,j}$ and $H_{i,j}$ are the width and height of the ith and jth patches, respectively.

Finally, the whole network architecture is trained with Adam optimizer [KB14] with a 10^{-4} learning rate. The overview of the whole architecture is shown in Figure 5.4

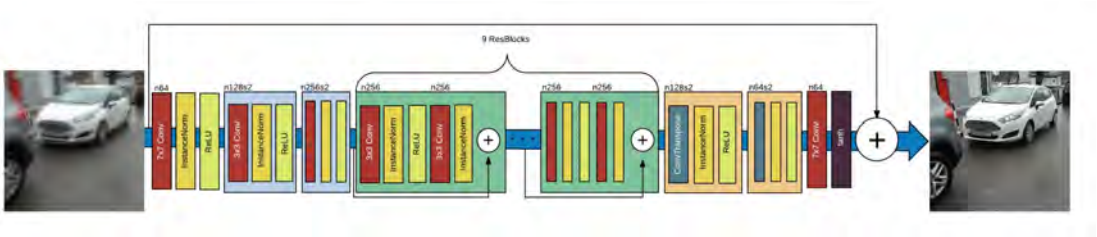


Figure 5.2: DeblurGAN generator is made of two strided convolution blocks, nine residual blocks [He+15], and two transposed convolution blocks. Each residual block has a convolution layer, instance normalization, and ReLU activation [Kup+18].

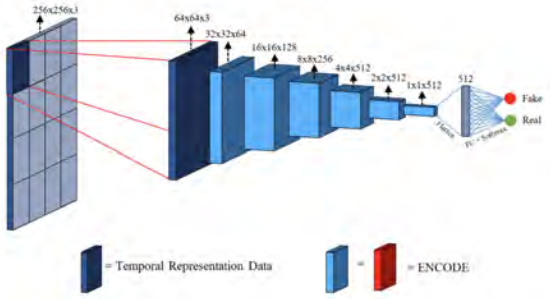


Figure 5.3: PatchGAN discriminator is a convolutional network that runs on $N * N$ image patches to classify if the patch is fake or real. [GAS20].

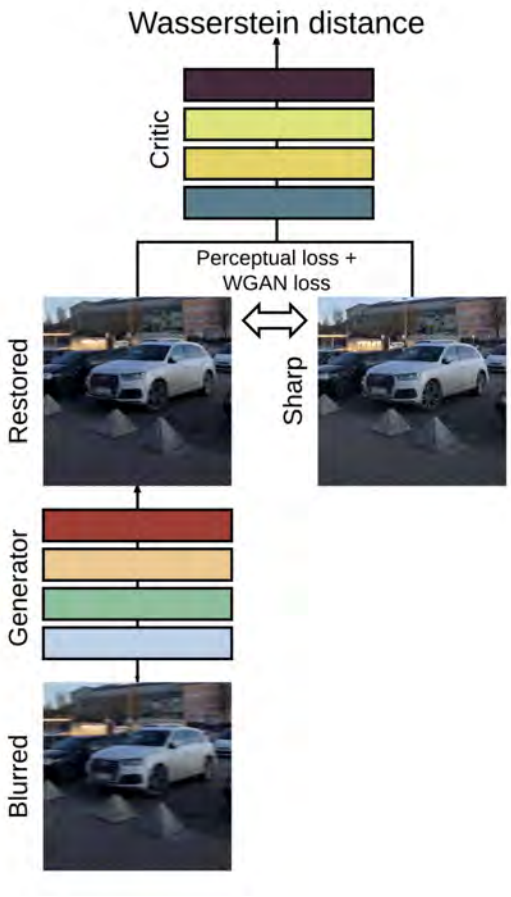


Figure 5.4: The overview of the DeblurGAN architecture [Kup+18]. In our case, the generator takes a blurred image, the fake SR image ,as input and produces a restored image, SR. The critic (discriminator) network takes both SR and HR images and outputs a distance between these two images. Total loss is calculated as the sum of the WGAN loss and perceptual loss[JAF16]. Perceptual loss is computed as the difference between the VGG-19's *conv*3.3 feature maps [SZ14] of the SR and HR images.

5.2 DeblurGANv2

After successfully implementing ultrasound super-resolution by utilizing Bicubic upscaling and DeblurGAN to fine-tune the super-resolved image, to achieve a better result, DeblurGANv2 is replaced with DeblurGAN in this pipeline which is an improved version of DeblurGAN [Kup+19], which is used in our experiments.

DeblurGANv2 is also an end-to-end GAN, increasing efficiency in inference time, image quality, and flexibility. DeblurGANv2 employs a Feature Pyramid Network (FPN) in the generator, which was originally developed for object detection and implemented for image reconstruction for the first time; it can be connected to different backbones such as Inception-ResNet-v2 [Sze+16] for better image deblurring quality, and MobileNet [San+18] for faster inference [Kup+19]. For the discriminator part, DeblurGANv2 employs a relativistic [Jol18] double-scale discriminator with a least-square loss [Mao+16], which evaluates global (image) and local (patch) scales.

By taking the same minimax game used in DeblurGAN:

$$\min_G \max_D V(D,G) = \mathop{\mathbb{E}}_{x\sim_{data}(x)}[\log D(x)] + \mathop{\mathbb{E}}_{z\sim p(z)}[\log(1-D(G(z)))] \tag{5.6}$$

Again, to deal with the optimization problems of this objective function, such as mode collapse and gradient vanishing/explosion, Least Squares GAN loss [Mao+16] is used to generate a smoother and non-saturating gradient. Which removes log to deal with quick saturation, and by utilizing $L2$ loss, fake samples get larger penalties. Also, the proposed loss function minimizes the Pearson χ^2 divergence, which improves the training stability [Kup+19]. Therefore the loss function is defined as:

$$\begin{aligned} \min_D V(D) = \frac{1}{2}\mathop{\mathbb{E}}_{x\sim p_{data}(x)}[(D(x)-1)^2] + \frac{1}{2}\mathop{\mathbb{E}}_{z\sim p(z)}[D(G(z))^2] \\ \min_G V(G) = \frac{1}{2}\mathop{\mathbb{E}}_{z\sim p(z)}[(D(G(z))-1)^2] \end{aligned} \tag{5.7}$$

Then instead of using WGAN-GP discriminator in DeblurGAN [Kup+18], authors [Kup+19] proposed to use a relativistic wrapping [Jol18] on top of the LSGAN [Mao+16] cost function which results RaGAN-LS loss:

$$\begin{aligned} L_D^{RaLSGAN} = \mathop{\mathbb{E}}_{x\sim p_{data}(x)}[(D(x) - \mathop{\mathbb{E}}_{z\sim p(z)} D(G(z)) - 1)^2] \\ + \mathop{\mathbb{E}}_{z\sim p(z)}[(D(G(z)) - \mathop{\mathbb{E}}_{x\sim p_{data}(x)} D(x) + 1)^2] \end{aligned} \tag{5.8}$$

Compared to WGAN-GP loss, it is reported that RaGAN-LS loss is more stable and faster to train, produces higher perceptual quality, and generates sharper images [Kup+19]. The overview of the DeblurGANv2 can be seen in Figure 5.5.

For the overall loss, it is defined as:

$$L_G = 0.5 * L_p + 0.006 * L_X + 0.01 * L_{adv} \tag{5.9}$$

where L_p is the pixel space loss such as L_1 or L_2 loss, where MSE loss is chosen by design choice to help correct color and texture distortions [Kup+19]. L_X is the content loss which uses VGG19 [SZ14] *conv*3.3 feature maps, and L_{adv} is the adversarial loss, which covers the discriminator losses.

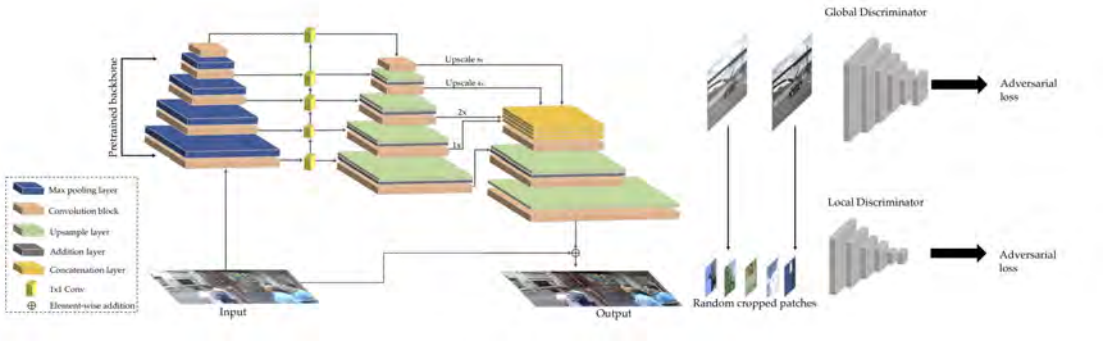


Figure 5.5: Overview of the DeblurGANv2. As for the generator uses FPN to cover different scales of features, and for the discriminator, it uses a double-scale discriminator to cover both patch and image-level classification. [Kup+19].

In addition to all of the changes proposed in DeblurGANv2 [Kup+18], we implemented a pre-trained Swin Transformer v2 architecture shown in Figure 5.6 and the key changes shown in the Figure 5.7 [Liu+21b] into the FPN to have the benefits of using Transformer.

Different than Inception-ResNet-v2 [Sze+16] pre-trained Swin v2 [Liu+21b] only has 4 stages; therefore, we connected 4 of the FPN layers to the Swin Transformer v2's features [Liu+21b] by disabling the biggest feature map, with respectively upscaled to 8, 4, 2 and 1 concatenated, fed into a convolution to smoothen then all of them are upscaled by 2 with nearest neighbor (NN) mode. Also, to generate the biggest map, the first map from Swin v2 is bicubically upscaled by 2 and again fed into another smoothing convolution. Finally, they are upscaled again 2 with NN mode.

5.3 Nonlinear Activation Free Network for Image Restoration

Finally, we decided to look at one of the recent works in the image deblurring area for ultrasound super-resolution by utilizing deblurring tools to improve the results even further. Nonlinear Activation Free Network (NAFNet) [Che+22a] tries to achieve state-of-the-art (SOTA) by having a low inter-block complexity by utilizing single-stage UNet

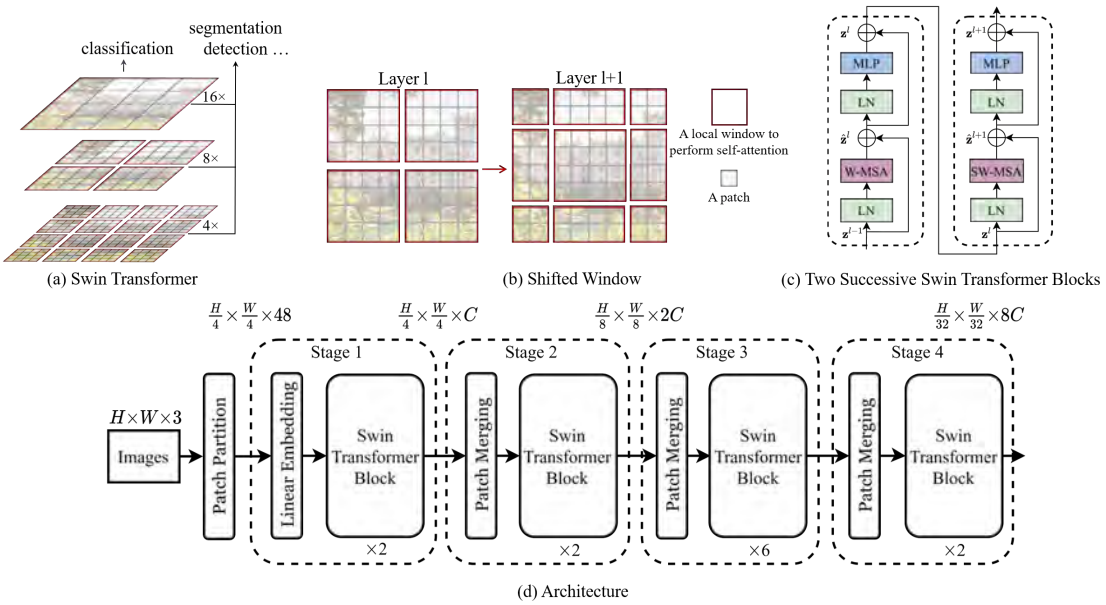


Figure 5.6: Architecture of Swin Transformer v2, where it consists of 4 stages. [Liu+21b]

which shown in Figure 5.8 [RFB15] and low intra-block complexity, which starts with the simplest block design such as convolution, ReLU activation and shortcut [He+15] which shown in Figure 5.11b. Then by experimenting with different components, authors created a baseline model which contains layer normalization, convolution, Gaussian Error Linear Unit (GELU) [HG16], which can be seen in Figure 5.9 and Channel Attention Module (CA), which is computationally efficient and brings global information to the feature map, where self-attention suffers from complexity and fix-sized local window self-attention suffers from the global information [Che+22a]. The baseline block is shown in Figure 5.11c.

By pushing the simplicity even further, authors created NAFNet by checking the two concepts: Simplifying the Gated Linear Units and the CA block. Where gated linear units [Dau+16] can be formulated as:

$$Gate(\mathbf{X}, f, g, \sigma) = f(\mathbf{X} \odot \sigma(g(\mathbf{X}))) \tag{5.10}$$

where $\mathbf{X}$ is the feature map, f and g are the linear transformers, σ is a nonlinear activation function and $\odot$ is the element-wise multiplication. Since GLU [Dau+16] increases the intra-block complexity, authors decided to use GELU [HG16]:

$$GELU(x) = x\Phi(x) \tag{5.11}$$

where Φ represents the cumulative distribution function for the standard normal

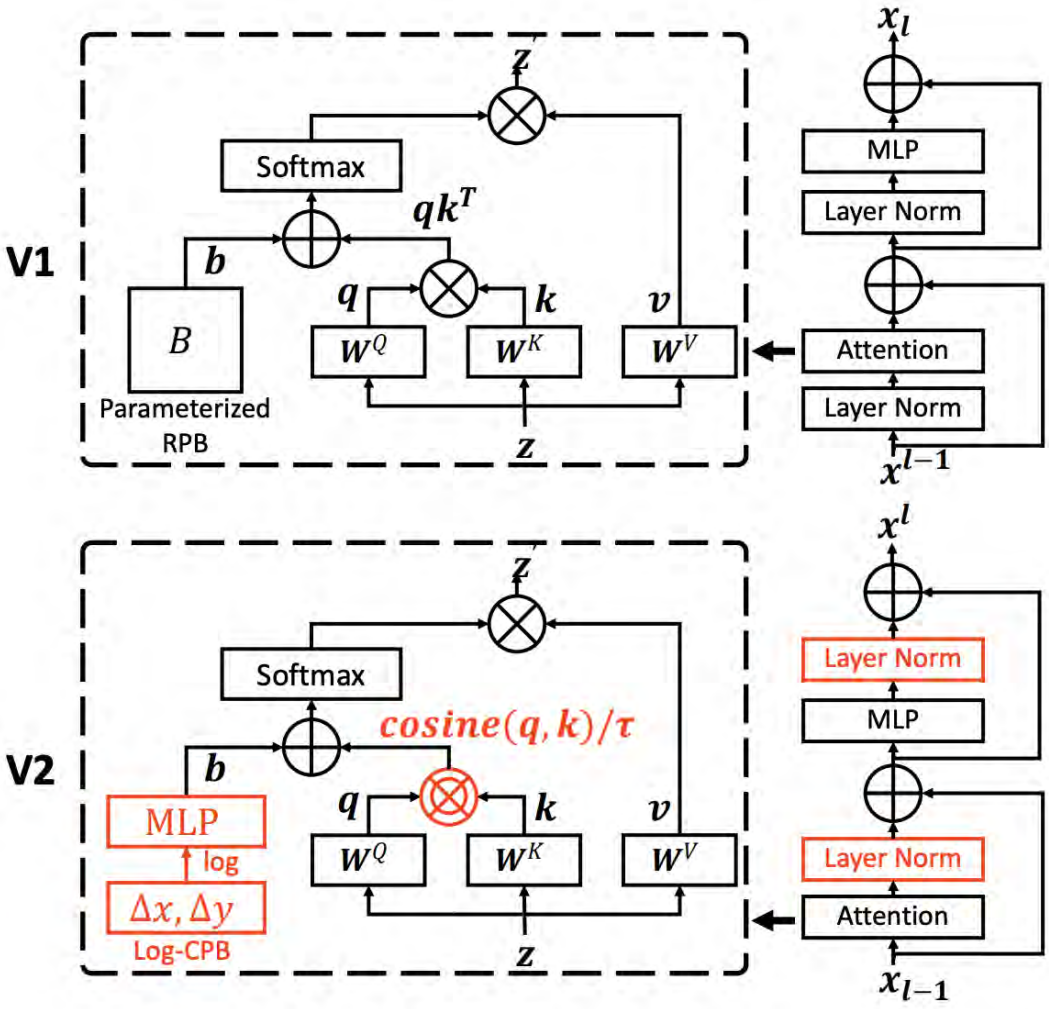


Figure 5.7: Architecture of Swin Transformer v2, where it changes the order of layer norms with a post-normalization and adds scaled cosine on the attention function instead of the dot product to make it easier to scale up the capacity and also replaces parameterized relative position bias with log-spaced continuous RPB to transfer model more effectively across window resolutions. [Liu+21b]

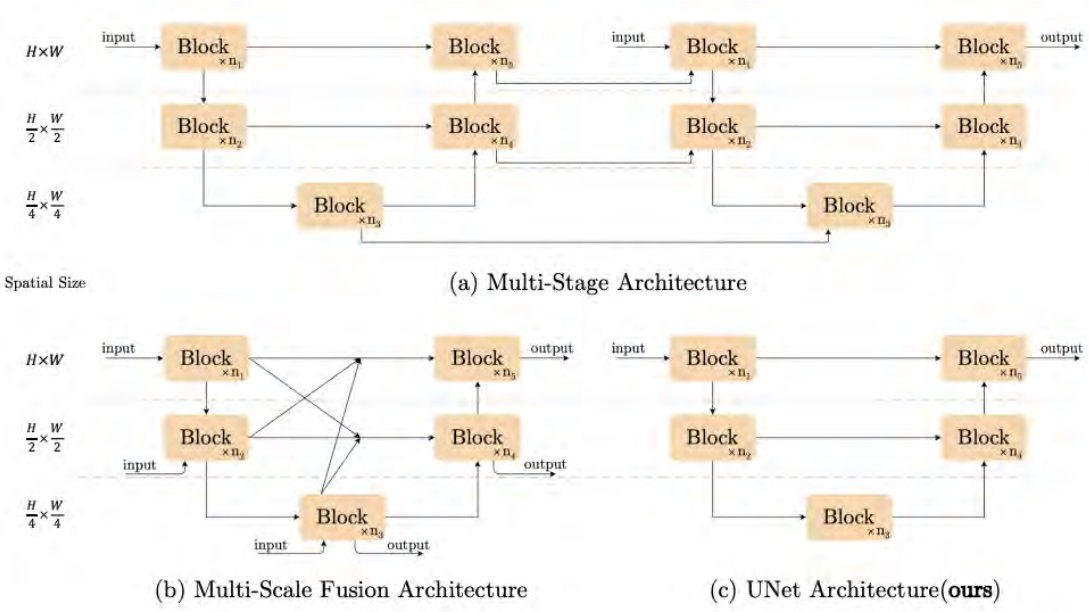


Figure 5.8: Overview of the image restoration model architectures. (a) The Multi-Stage Architecture is just a stacked UNet. (b) The Multi-Scale Fusion Architecture is a UNet with skip connections from different scales. (c) Finally the architecture of UNet [RFB15], which is the simplest image restoration architecture, used in NAFnet [Che+22a].

distribution, which is implemented as:

$$\Phi(x) = \frac{1}{2}x(1 + \tanh(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)]) \tag{5.12}$$

therefore, GELU is a special case which σ is ϕ and f and g are identity functions. Therefore, by taking nonlinearity out, similar to GELU, authors [Che+22a] propose the SimpleGate function:

$$SimpleGate(\mathbf{X}, \mathbf{Y}) = \mathbf{X} \odot \mathbf{Y} \tag{5.13}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are feature maps of same size and $\odot$ is the element-wise multiplication. The only nonlinear activations left in the baseline are Sigmoid and ReLU in the CA, which is modeled as:

$$CA(\mathbf{X}) = \mathbf{X} * \sigma(W_2 \max(0, W_1 \text{pool}(\mathbf{X}))) \tag{5.14}$$

where $\mathbf{X}$ denotes the feature map, $*$ is the channelwise product operation, pool is global average pooling and finally σ is sigmoid activation and max operation denotes ReLU activation between W_1, W_2 layers. Therefore, by eliminating the activation functions to remove nonlinearity, the Simplified Channel Attention [Che+22a] is proposed as:

$$SCA(\mathbf{X}) = \mathbf{X} * W\text{pool}(\mathbf{X}) \tag{5.15}$$

.

Finally, the changes shown in Figure 5.10 are applied to the baseline block, and the final block is shown in Figure 5.11d.

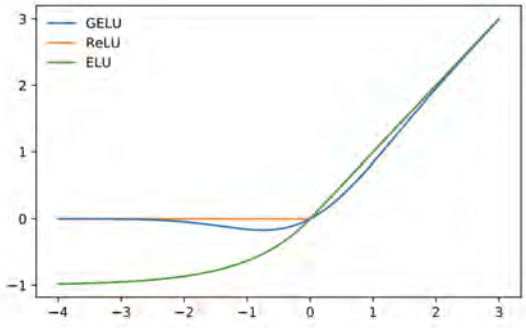


Figure 5.9: Comparison of the GELU [HG16] activation function with ReLU and ELU [CUH15] activations.

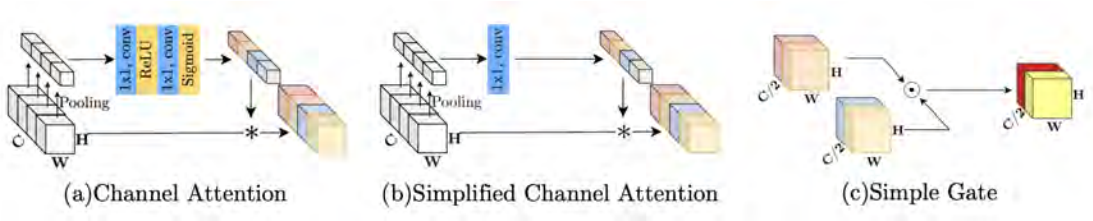


Figure 5.10: (a) Channel attention [Hu+17] model is simplified as (b)Simplified Channel Attention, and GELU activation is replaced with (c) Simple Gate. [Che+22a]

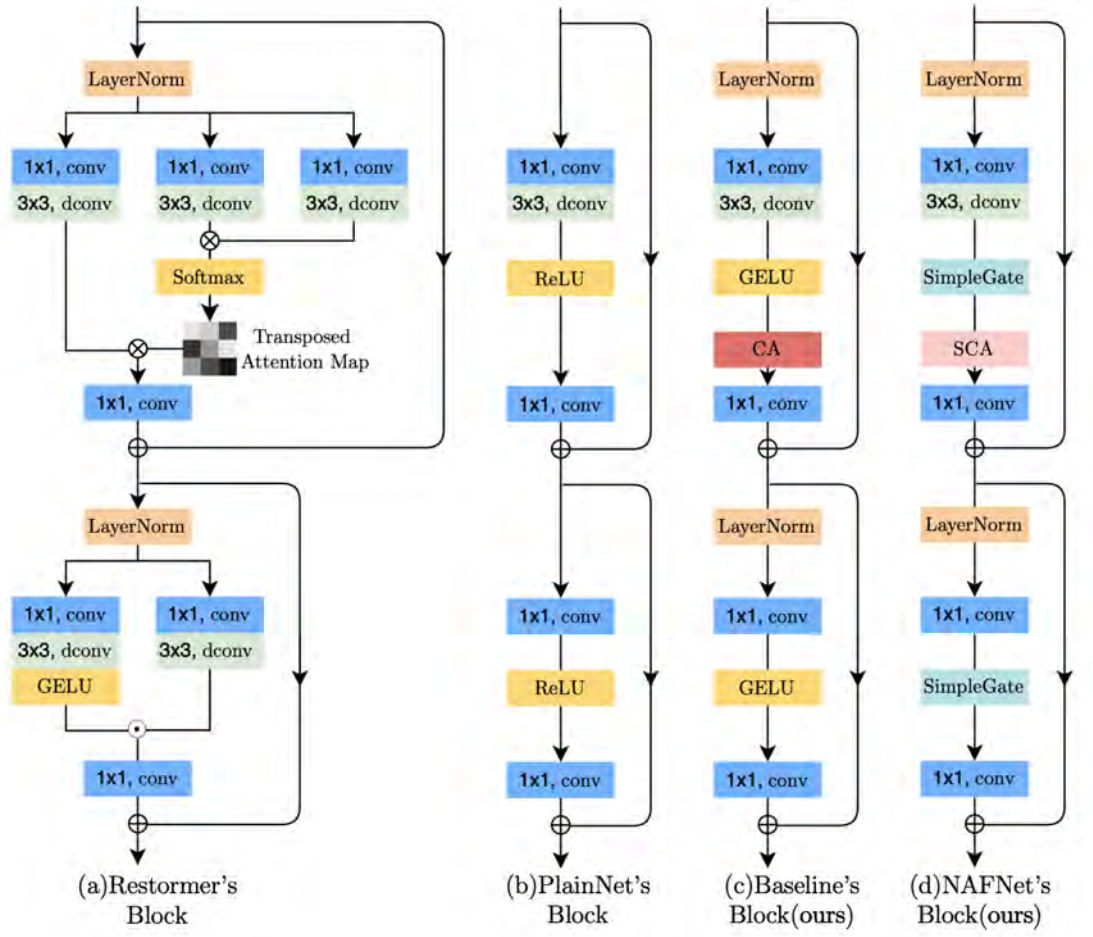


Figure 5.11: Different architectures for intra-blocks. (a) is Restormer's block [Zam+21], which is another efficient image restoration network with some details omitted, (b) is PlainNet's block, (c) is NAFNet's Baseline block, and finally, (d) is the NAFNet's block. [Che+22a].

5.4 HAT - NAF Mixture

As another proposal, we decided to combine the best of both worlds, HAT [Che+22b] and NAFNet [Che+22a], to see if we can achieve better results. HAT is a SOTA image restoration network which utilizes CA [Hu+17], therefore we decided to use NAFNet's Simplified Channel Attention (SCA) [Che+22a] instead of CA [Hu+17] in HAT [Che+22b] and also, we removed the activation functions such as ReLU in the upsampler. Also, we changed all activation functions in CA block with SimpleGate of NAFNet [Che+22a] since we replace the whole CA block with SCA to see if we can achieve better results. The final architecture is shown in Figure 5.12.

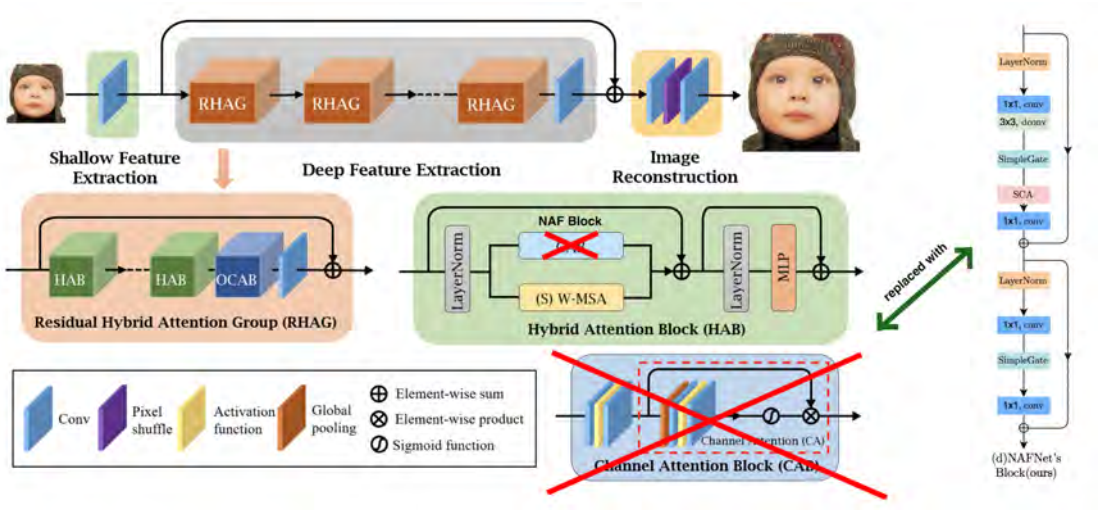


Figure 5.12: The architecture of HAT-NAF. Basically CA block of HAT [Che+22b] is replaced with NAFNet's NAF block [Che+22a].

6 Experiments & Results

In this section, the settings of the experiments are presented. Then, the results of these experiments are presented. First, the results of the experiments on the CCA-US dataset are presented. Then, the results of the experiments on the BUSI dataset are presented. Each of these datasets is benchmarked in two different categories, no-blur and Gaussian blurred.

6.1 Experiments

For this work, we have trained various SR and deblurring networks such as Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) [Wan+18], EDSR [Lim+17], SwinIR [Lia+21], HAT [Che+22b], DeblurGANv2 [Kup+19] and NAFNet [Che+22a], our proposed SR and deblurring networks and our proposed NAF-HAT mixture network and tested on these networks where PSNR for validation is highest for each network respectively and PRLMU [Liu+22] on the CCA-US and BUSI datasets. In the following subsubsections, the settings of the experiments are presented.

6.1.1 ESRGAN

For training the ESRGAN network, we have used the ESRGAN implementation from BasicSR [Wan+22] and trained the network on the CCA-US and BUSI datasets. The ESRGAN network is trained with the 4x SR setting from $64 * 64$ to $256 * 256$. It has trained on two NVIDIA TITAN Xp with a batch size of 8. It utilizes the RRDB [Wan+18] network with 23 residual blocks and 64 features. It has trained with the Adam optimizer with a learning rate of 10^{-4} and ran for 10000 iterations with L1 loss with the weight of 10^{-2}, Perceptual VGG, and vanilla GAN loss with the weight of $5 * 10^{-3}$.

6.1.2 EDSR

For training the EDSR network, we have used the EDSR implementation from BasicSR [Wan+22] and trained the network on the CCA-US and BUSI datasets with and without blur degraded configurations. The EDSR network is trained with the 4x SR setting from $64 * 64$ to $256 * 256$. It has trained on one NVIDIA RTX A5000 with a batch size of 16. The EDSR network contains 32 residual blocks with 256 features. It has trained with the Adam optimizer with a learning rate of 10^{-4} and ran for 10000 iterations with L1 loss.

6.1.3 SwinIR

For training the SwinIR network, we have used the SwinIR implementation provided by BasicSR [Wan+22] and trained the network on the CCA-US and BUSI datasets with and without blur degraded configurations. The SwinIR network is also trained with the 4x SR setting from $64 * 64$ to $256 * 256$. It has trained on two NVIDIA RTX A5000 GPUs with a batch size of 4 per GPU. The SwinIR network has an input image size of $48 * 48$ and a window size of 8. It has 6 stages with 6 layers per stage and 6 attention heads for each layer with an embedded dimension of 180 with an MLP ratio of 2, and as the upsampler, pixel shuffle is used. It has trained with the Adam optimizer with a learning rate of $2 * 10^{-4}$ and multi-step scheduler gamma of 0.5 with milestones at $5000, 8000, 9000, 9500$ and ran for a total of 10000 iterations with L1 loss.

6.1.4 HAT and NAF-HAT Mixture

For training the HAT network, we have used the HAT implementation by the authors of HAT [Che+22b]. The HAT network is trained with the 4x SR setting from $64 * 64$ to $256 * 256$. It has trained on two NVIDIA RTX A5000 GPUs with a batch size of 2 per GPU. It takes $64 * 64$ input image size with a window size of 16, compress ratio of 3, and squeeze factor of 30. It has 12 stages with 6 layers per stage and 6 attention heads for each layer with an embedded dimension of 180 with an MLP ratio of 2, and as the upsampler, pixel shuffle is used, similar to SwinIR [Lia+21]. It has trained with the Adam optimizer with a learning rate of $2 * 10^{-4}$ and multi-step scheduler gamma of 0.5 with milestones at $3000, 5000, 6500, 7000, 7500$ and ran for a total of 10000 iterations with L1 loss.

6.1.5 PRLMU

Since PRLMU [Liu+22] does not have any training methods supplied by authors, we only have tested the PRLMU network supplied by the authors on the CCA-US and BUSI datasets with and without blur degraded configurations.

6.1.6 DeblurGANv2

For training the DeblurGANv2 [Kup+19] network, we have used the DeblurGANv2 implementation from the authors [Kup+19] and trained the network on the CCA-US and BUSI datasets with and without blur degraded configurations. The DeblurGANv2 network is trained with the bicubically upsampled images of $256 * 256$ to high-resolution images with the same $256 * 256$ resolution. It has trained on two NVIDIA RTX A5000 GPUs with a batch size of 4 per GPU. We have trained two different models of the DeblurGANv2 network; one employs the Inception ResNet v2 [Sze+16] as FPN, and the other one is our proposed with Swin v2 [Liu+21b] as FPN. Also, we have trained two different models of the DeblurGANv2 network; one employs the Inception ResNet

v2 [Sze+16] as FPN, and the other one is our proposed with Swin v2 [Liu+21b] as FPN, these networks marked as "Aug", they use bicubically upsampled datasets but with augmentation that is used in DeblurGANv2's original implementation, such as motion blur, median blur, gamma, RGB and HSV shifts, sharpening, JPEG distortion, cutouts [Kup+19].

6.1.7 NAFNet

For training the NAFNet network, we have used the original NAFNet implementation provided by the authors [Che+22a]. The NAFNet network is trained with the 1x "refinement" setting in $256 * 256$ where the input image is either blurred and bicubically upsampled or only bicubically upsampled image from the CCA-US and BUSI datasets. It has trained on two NVIDIA RTX A5000 GPUs with a batch size of 8 per GPU. It is trained on the network type "NAFNetLocal", which enables Test-time Local Conversion [Chu+22] with a width of 64 and encoder blocks of $1, 1, 1, 28$, one middle block and decoder blocks of $1, 1, 1, 1$. It has trained with the Adam optimizer with a learning rate of 10^{-3} with $10^{-}3$ weight decay, $0.9, 0.9$ betas, cosine annealing LR scheduler with minimum LR of $10^{-}7$ and ran for 10000 iterations with PSNR loss.

6.2 Evaluation Metrics

In this section, we present the evaluation metrics used in the experiments. The evaluation metrics used in the experiments are Peak Signal-to-Noise Ratio (psnr) and Structural Similarity Index (SSIM) [HZ10]. The PSNR is a measure of the quality of the reconstructed image, and it is defined as:

$$\text{PSNR} = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \tag{6.1}$$

where MAX_I is the maximum valid value for a pixel such that 1 or 255 and MSE is the mean squared error between the original and the reconstructed image such that:

$$MSE = \frac{1}{c * i * j} \sum (I_1 - I_2)^2 \tag{6.2}$$

where i and j are the height and width of the image, respectively, and c is the number of channels of the images I_1 and I_2.

The SSIM is a measure of the similarity between the original and the reconstructed image, and it is defined as:

$$\text{SSIM} = l(x, y) \cdot c(x, y) \cdot s(x, y) \tag{6.3}$$

where $l(x, y)$ is the luminance component, $c(x, y)$ is the contrast component, and $s(x, y)$ is the structure component. The luminance component is defined as:

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{6.4}$$

where μ_x and μ_y are the mean values of the original and the reconstructed image, respectively, and C_1 is a constant that is used to stabilize the division by zero. The value of C_1 is chosen as $C_1 = (K_1 L)^2$ where L is the dynamic range of the pixel values and $K_1 << 1$ is a small constant. The contrast component is defined as:

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{6.5}$$

where σ_x and σ_y are the standard deviations of the original and the reconstructed image, respectively, and C_2 is a constant that is used to stabilize the division by zero. The value of C_2 is chosen as $C_2 = (K_2 L)^2$ where L is the dynamic range of the pixel values and $K_2 << 1$ is a small constant. The structure component is defined as:

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{6.6}$$

where σ_{xy} is the covariance between the original and the reconstructed image, and C_3 is a constant that is used to stabilize the division by zero. The SSIM is calculated for each channel of the image separately and then averaged over all channels. The SSIM is calculated for each channel of the image separately and then averaged over all channels.

6.3 Results

In the following sections, we present the results of the experiments. The results of the experiments are presented for CCA with bicubic degradation in Table 6.1, for CCA with blur + bicubic degradation in Table 6.2, for BUSI with bicubic degradation in Table 6.3 and for BUSI with blur + bicubic degradation in Table 6.4. The results are ordered by PSNR in each table. Then the results of the experiments are presented as figures in Figure 6.1, Figure 6.2, Figure 6.3 and Figure 6.4, where the figures contain the images respectively from the same datasets (there will be no cross-overs between BUSI and CCA) and blurred images are reconstructed from 3.0 sigma blur kernel over the gaussian blur size of 21. Also, the PSNR and SSIM values reported on the figures are not representing the single image but indicate the average value of the test set (i.e., taken from the tables for easier comparison).

6.4 Achieving Real Super-Resolution

After training these models, we can achieve real super-resolution by feeding HR images as the LR image of the model. The results are presented in Figure 6.5 and

Model	Dataset Trained On	PSNR	SSIM
HATNAF	CCA	35.7466	0.9343
HAT	CCA	35.4987	0.9329
HAT	BUSI	34.9785	0.9285
HATNAF	BUSI	34.8797	0.9245
NAFNet	CCA	33.7691	0.9055
EDSR	CCA	33.6931	0.9071
SwinIR	CCA	33.5828	0.9107
SwinIR	BUSI	33.2998	0.9057
NAFNet	CCABLUR	33.1674	0.8925
EDSR	BUSI	33.0896	0.8714
NAFNet	BUSI	33.0192	0.8857
NAFNet	BUSIBLUR	32.9580	0.8910
DeblurGANv2-Inception-Aug	CCA	32.8221	0.8112
HAT	CCABLUR	32.6455	0.9172
DeblurGANv2-Inception	BUSI	32.6343	0.8346
DeblurGANv2-Swinv2	BUSI	32.6076	0.8258
DeblurGANv2-Swinv2-Aug	BUSI	32.6014	0.8308
DeblurGANv2-Inception-Aug	BUSI	32.4752	0.8274
DeblurGANv2-Swinv2	CCA	32.4706	0.7153
DeblurGANv2-Inception	CCA	32.4531	0.8159
DeblurGANv2-Inception	CCABLUR	32.3528	0.8113
HAT-NAF	CCABLUR	32.2511	0.9122
DeblurGANv2-Swinv2-Aug	CCA	32.1360	0.7908
SwinIR	CCABLUR	31.7237	0.8885
DeblurGANv2-Inception	BUSIBLUR	31.4022	0.8179
HAT	BUSIBLUR	31.1572	0.9055
DeblurGANv2-Swinv2	BUSIBLUR	30.7294	0.7948
SwinIR	BUSIBLUR	30.3656	0.8887
DeblurGANv2-Swinv2	CCABLUR	30.2515	0.8017
HAT-NAF	BUSIBLUR	30.1834	0.8883
ESRGAN	CCA	30.0244	0.8153
ESRGAN	BUSI	29.6859	0.8328
EDSR	BUSIBLUR	29.2012	0.8631
EDSR	CCABLUR	28.9379	0.8402
PRLMU-pretrained	CCABLUR	27.25086	0.830039
ESRGAN	CCABLUR	26.1603	0.7303
ESRGAN	BUSIBLUR	19.0914	0.5315

Table 6.1: Results of the CCA Dataset

Model	Dataset Trained On	PSNR	SSIM
HAT	CCABLUR	35.3914	0.9279
HATNAF	CCABLUR	35.2719	0.9273
HATNAF	BUSIBLUR	34.866	0.9222
HAT	BUSIBLUR	34.8072	0.9233
PRLMU-pretrained	CCABLUR	33.6734	0.8984
SwinIR	CCABLUR	33.3035	0.8898
EDSR	CCABLUR	33.1699	0.8924
SwinIR	BUSIBLUR	33.1528	0.9006
EDSR	BUSIBLUR	33.0655	0.8993
NAFNet	CCABLUR	32.6682	0.8772
NAFNet	BUSIBLUR	32.3801	0.8718
DeblurGANv2-Swinv2	BUSIBLUR	31.4899	0.781
DeblurGANv2-Swinv2	CCABLUR	31.4514	0.7802
HATNAF	CCA	30.5964	0.8689
DeblurGANv2-Inception	CCABLUR	30.4892	0.7739
HATNAF	BUSI	30.3944	0.8652
HAT	CCA	30.3591	0.8639
HAT	BUSI	30.3379	0.8643
DeblurGANv2-Inception	BUSIBLUR	30.2932	0.7762
ESRGAN	CCABLUR	29.9351	0.8026
NAFNet	CCA	29.2018	0.8242
SwinIR	BUSI	29.1363	0.8282
NAFNet	BUSI	29.1045	0.8173
EDSR	CCA	29.0919	0.8183
SwinIR	CCA	29.0329	0.825
EDSR	BUSI	28.9884	0.7862
ESRGAN	BUSIBLUR	28.9588	0.7913
ESRGAN	BUSI	28.8838	0.8107
DeblurGANv2-Inception	CCA	28.8636	0.7484
DeblurGANv2-Inception-Aug	CCA	28.8597	0.7457
ESRGAN	CCA	28.8185	0.7804
DeblurGANv2-Swinv2	BUSI	28.7989	0.7612
DeblurGANv2-Swinv2	CCA	28.7466	0.7153
DeblurGANv2-Swinv2-Aug	CCA	28.6449	0.7179
DeblurGANv2-Inception	BUSI	28.6429	0.7612
DeblurGANv2-Swinv2-Aug	BUSI	28.5809	0.7533
DeblurGANv2-Inception-Aug	BUSI	28.5729	0.758

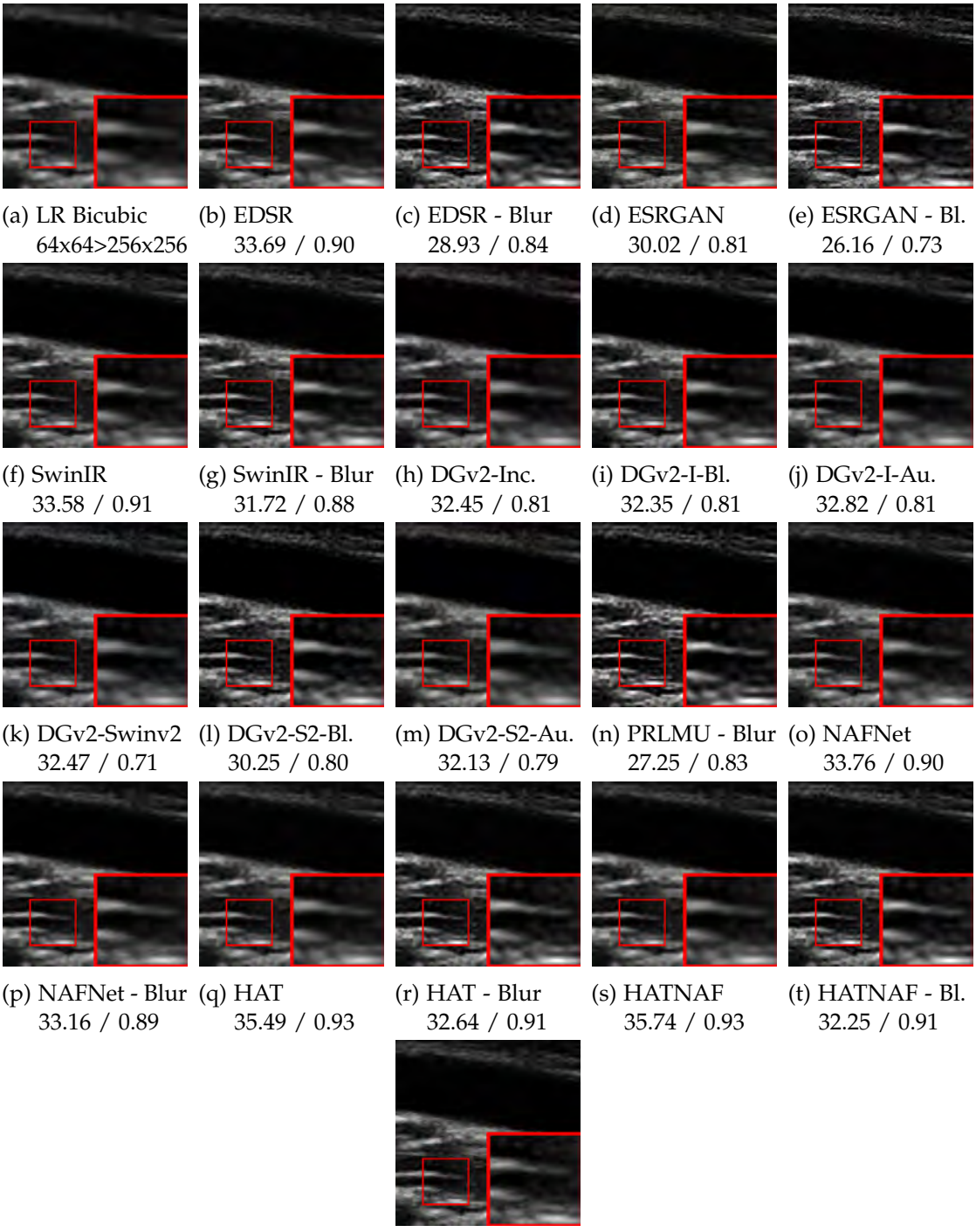
Table 6.2: Results of the CCA-Blur Dataset

Model	Dataset Trained On	PSNR	SSIM
HAT	BUSI	32.474	0.8715
HATNAF	CCA	32.2632	0.8649
HATNAF	BUSI	32.0637	0.8615
HAT	CCA	32.0081	0.86
SwinIR	BUSI	30.7687	0.8417
EDSR	BUSI	30.6664	0.8398
SwinIR	CCA	30.3294	0.8277
DeblurGANv2-Swinv2-Aug	BUSI	29.784	0.9297
NAFNet	BUSI	29.7682	0.7886
HATNAF	CCABLUR	29.6303	0.8213
NAFNet	BUSIBLUR	29.619	0.7942
DeblurGANv2-Swinv2	BUSI	29.5974	0.928
EDSR	CCA	29.5318	0.7988
HAT	CCABLUR	29.5039	0.8158
DeblurGANv2-Inception	BUSI	29.4538	0.9273
NAFNet	CCA	29.4246	0.7786
DeblurGANv2-Swinv2	CCA	29.297	0.9286
DeblurGANv2-Inception-Aug	BUSI	29.2466	0.9263
HAT	BUSIBLUR	29.2198	0.8186
DeblurGANv2-Inception	CCA	29.1303	0.9287
DeblurGANv2-Swinv2-Aug	CCA	28.7398	0.923
DeblurGANv2-Inception-Aug	CCA	28.7336	0.9248
SwinIR	BUSIBLUR	28.6349	0.8212
HATNAF	BUSIBLUR	28.5693	0.8061
SwinIR	CCABLUR	28.4765	0.7932
DeblurGANv2-Inception	BUSIBLUR	28.4501	0.9288
NAFNet	CCABLUR	28.3813	0.7531
DeblurGANv2-Swinv2	BUSIBLUR	27.9851	0.9116
DeblurGANv2-Swinv2	CCABLUR	27.8203	0.9094
DeblurGANv2-Inception	CCABLUR	27.4287	0.9093
EDSR	BUSIBLUR	27.0586	0.7762
EDSR	CCABLUR	26.0705	0.7135
ESRGAN	CCA	26.0595	0.6538
ESRGAN	BUSI	25.9494	0.7082
PRLMU-pretrained	CCABLUR	25.257134	0.738353
ESRGAN	CCABLUR	22.7761	0.5255
ESRGAN	BUSIBLUR	19.5955	0.4465

Table 6.3: Results of the BUSI Dataset

Model	Dataset Trained On	PSNR	SSIM
HAT	BUSIBLUR	32.1634	0.857
HATNAF	BUSIBLUR	32.0966	0.8555
HATNAF	CCABLUR	31.4532	0.8436
HAT	CCABLUR	31.377	0.8425
SwinIR	BUSIBLUR	30.6431	0.834
EDSR	BUSIBLUR	30.2703	0.8276
SwinIR	CCABLUR	29.528	0.8046
PRLMU-pretrained	CCABLUR	29.340317	0.81773
NAFNet	BUSIBLUR	29.1153	0.7646
DeblurGANv2-Swinv2	BUSIBLUR	28.8459	0.9174
HATNAF	CCA	28.7869	0.7709
HAT	BUSI	28.7283	0.7707
HATNAF	BUSI	28.7158	0.7689
EDSR	CCABLUR	28.6705	0.7759
HAT	CCA	28.6129	0.7651
DeblurGANv2-Swinv2	CCABLUR	28.372	0.9099
NAFNet	CCABLUR	27.8738	0.7297
DeblurGANv2-Inception	BUSIBLUR	27.5548	0.9066
DeblurGANv2-Inception	CCABLUR	27.4876	0.9045
SwinIR	BUSI	27.4599	0.7384
EDSR	BUSI	27.419	0.7355
SwinIR	CCA	27.3441	0.7334
NAFNet	BUSI	27.2842	0.7203
NAFNet	CCA	27.1005	0.7066
EDSR	CCA	27.0682	0.7123
DeblurGANv2-Swinv2	BUSI	26.9981	0.8715
DeblurGANv2-Swinv2-Aug	BUSI	26.9821	0.8686
DeblurGANv2-Inception	CCA	26.7742	0.8743
DeblurGANv2-Swinv2	CCA	26.7361	0.8707
DeblurGANv2-Inception	BUSI	26.6884	0.8665
DeblurGANv2-Inception-Aug	BUSI	26.5782	0.8654
ESRGAN	BUSIBLUR	26.7348	0.6864
DeblurGANv2-Swin-Aug	CCA	26.4426	0.8651
DeblurGANv2-Inception-Aug	CCA	26.1125	0.8643
ESRGAN	BUSI	25.983	0.7064
ESRGAN	CCA	25.9853	0.6751
ESRGAN	CCABLUR	26.0863	0.666

Table 6.4: Results of the BUSI-Blur Dataset



(a) LR Bicubic 64x64>256x256 | (b) EDSR 33.69 / 0.90 | (c) EDSR - Blur 28.93 / 0.84 | (d) ESRGAN 30.02 / 0.81 | (e) ESRGAN - Bl. 26.16 / 0.73

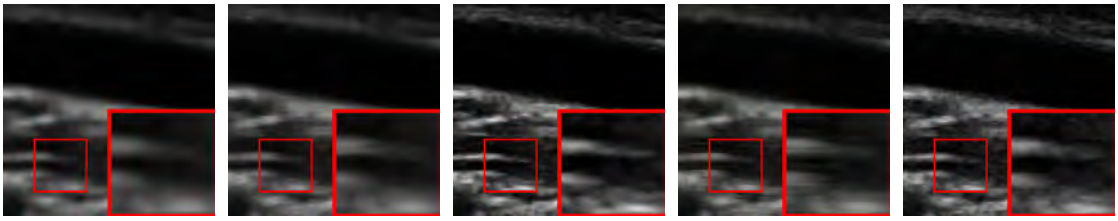
(f) SwinIR 33.58 / 0.91 | (g) SwinIR - Blur 31.72 / 0.88 | (h) DGv2-Inc. 32.45 / 0.81 | (i) DGv2-I-Bl. 32.35 / 0.81 | (j) DGv2-I-Au. 32.82 / 0.81

(k) DGv2-Swinv2 32.47 / 0.71 | (l) DGv2-S2-Bl. 30.25 / 0.80 | (m) DGv2-S2-Au. 32.13 / 0.79 | (n) PRLMU - Blur 27.25 / 0.83 | (o) NAFNet 33.76 / 0.90

(p) NAFNet - Blur 33.16 / 0.89 | (q) HAT 35.49 / 0.93 | (r) HAT - Blur 32.64 / 0.91 | (s) HATNAF 35.74 / 0.93 | (t) HATNAF - Bl. 32.25 / 0.91

(u) HR Image 256x256

Figure 6.1: CCA results



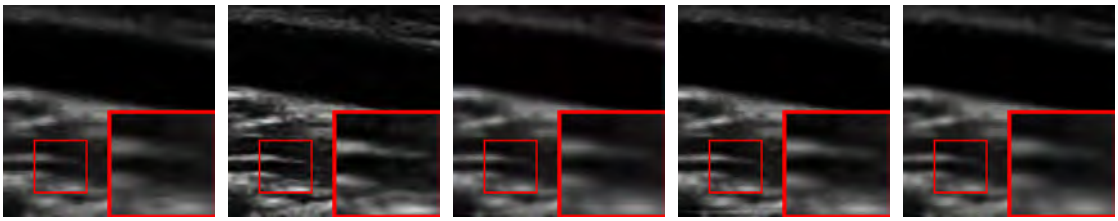
(a) LR Bicubic
64x64>256x256

(b) EDSR
29.09 / 0.81

(c) EDSR - Blur
33.16 / 0.89

(d) ESRGAN
28.81 / 0.78

(e) ESRGAN - Bl.
29.93 / 0.80



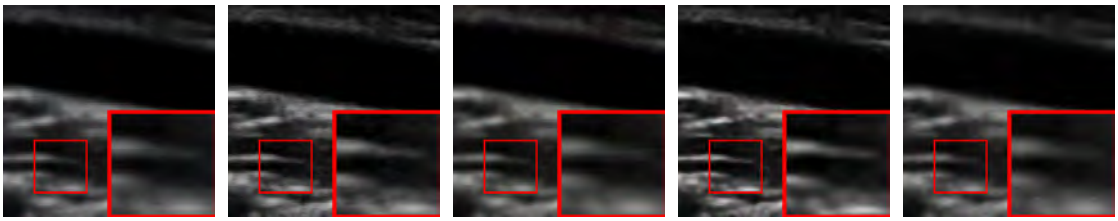
(f) SwinIR
29.03 / 0.82

(g) SwinIR - Blur
33.30 / 0.88

(h) DGv2-Inc.
28.86 / 0.74

(i) DGv2-I-Bl.
30.48 / 0.77

(j) DGv2-I-Au.
28.85 / 0.74



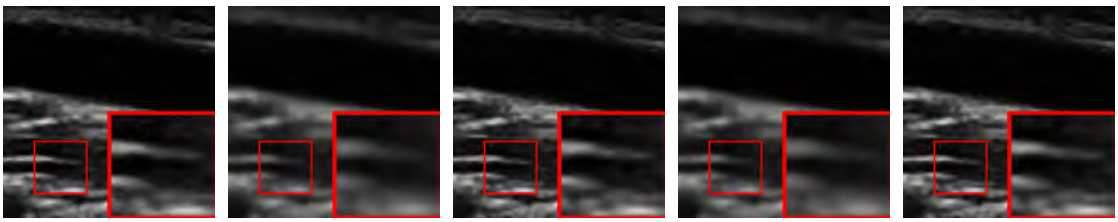
(k) DGv2-Swinv2
28.74 / 0.71

(l) DGv2-S2-Bl.
31.45 / 0.78

(m) DGv2-S2-Au.
28.64 / 0.71

(n) PRLMU - Blur
33.67 / 0.89

(o) NAFNet
29.20 / 0.82



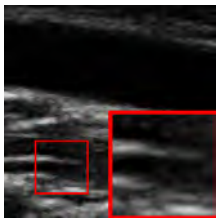
(p) NAFNet - Blur
32.66 / 0.87

(q) HAT
30.35 / 0.86

(r) HAT - Blur
35.39 / 0.92

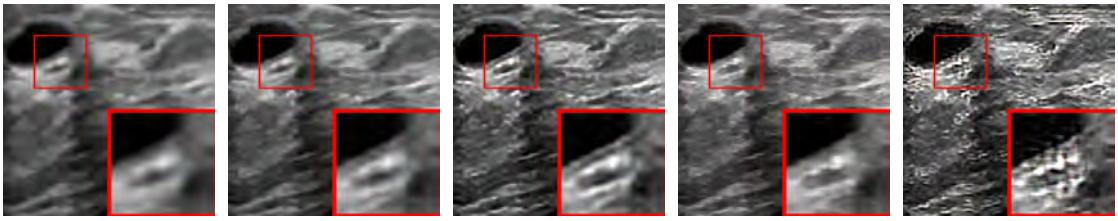
(s) HATNAF
30.59 / 0.86

(t) HATNAF - Bl.
35.27 / 0.92



(u) HR Image
256x256

Figure 6.2: CCA Blur results



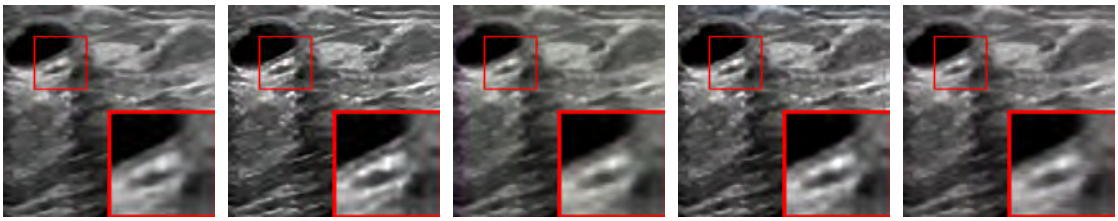
(a) LR Bicubic
64x64>256x256

(b) EDSR
30.66 / 0.83

(c) EDSR - Blur
27.05 / 0.77

(d) ESRGAN
25.94 / 0.70

(e) ESRGAN - Bl.
19.59 / 0.44



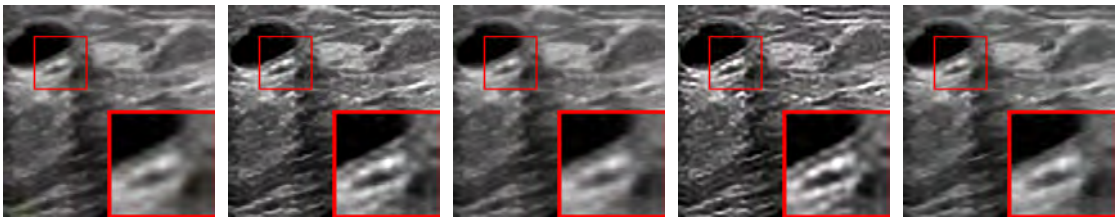
(f) SwinIR
30.76 / 0.84

(g) SwinIR - Blur
28.63 / 0.82

(h) DGv2-Inc.
29.45 / 0.92

(i) DGv2-I-Bl.
28.45 / 0.92

(j) DGv2-I-Au.
29.24 / 0.92



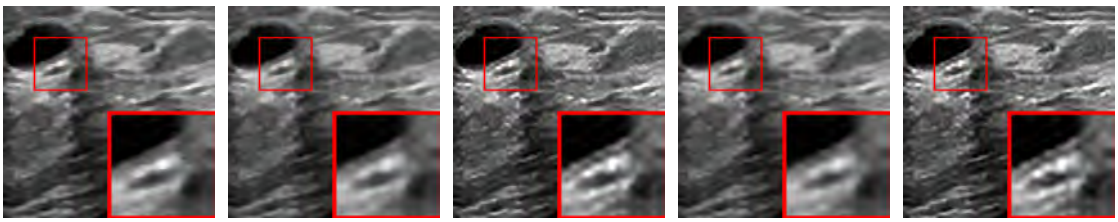
(k) DGv2-Swinv2
29.59 / 0.92

(l) DGv2-S2-Bl.
27.98 / 0.91

(m) DGv2-S2-Au.
29.78 / 0.92

(n) PRLMU - CBl.
25.25 / 0.73

(o) NAFNet
29.76 / 0.78



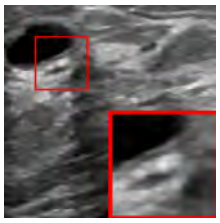
(p) NAFNet - Blur
29.61 / 0.79

(q) HAT
32.47 / 0.87

(r) HAT - Blur
29.21 / 0.81

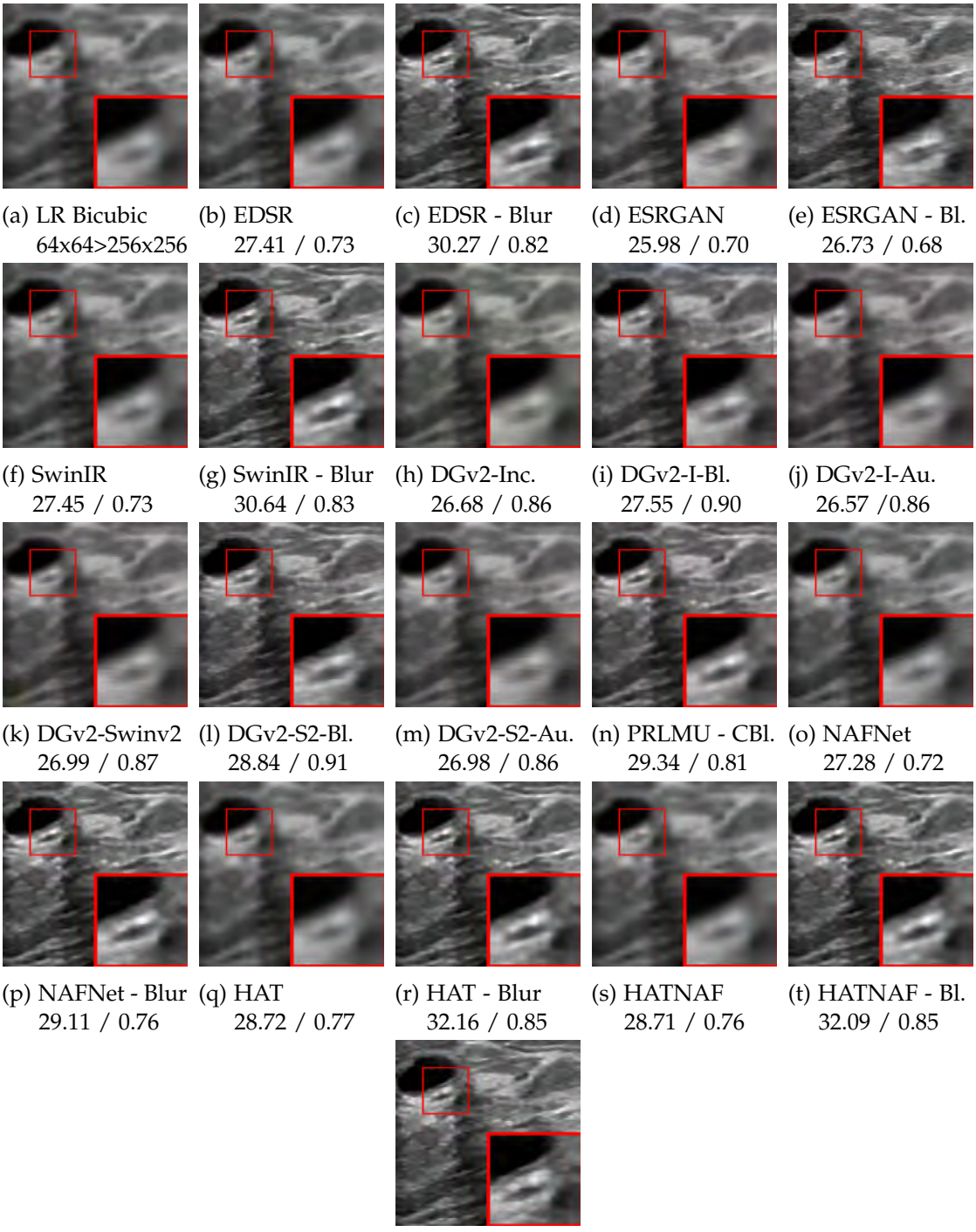
(s) HATNAF
32.06 / 0.86

(t) HATNAF - Bl.
28.56 / 0.80



(u) HR Image
256x256

Figure 6.3: BUSI results



(a) LR Bicubic 64x64>256x256 (b) EDSR 27.41 / 0.73 (c) EDSR - Blur 30.27 / 0.82 (d) ESRGAN 25.98 / 0.70 (e) ESRGAN - Bl. 26.73 / 0.68

(f) SwinIR 27.45 / 0.73 (g) SwinIR - Blur 30.64 / 0.83 (h) DGv2-Inc. 26.68 / 0.86 (i) DGv2-I-Bl. 27.55 / 0.90 (j) DGv2-I-Au. 26.57 /0.86

(k) DGv2-Swinv2 26.99 / 0.87 (l) DGv2-S2-Bl. 28.84 / 0.91 (m) DGv2-S2-Au. 26.98 / 0.86 (n) PRLMU - CBl. 29.34 / 0.81 (o) NAFNet 27.28 / 0.72

(p) NAFNet - Blur 29.11 / 0.76 (q) HAT 28.72 / 0.77 (r) HAT - Blur 32.16 / 0.85 (s) HATNAF 28.71 / 0.76 (t) HATNAF - Bl. 32.09 / 0.85

(u) HR Image 256x256

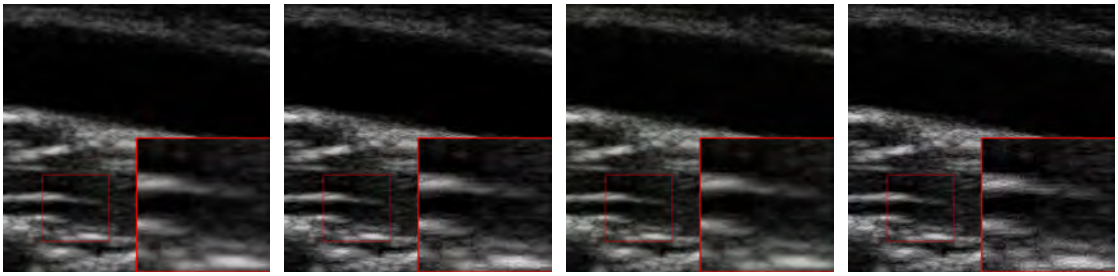
Figure 6.4: BUSI Blur results

Figure 6.6. Since it does not make sense to compare the PSNR and SSIM of the actual super-resolution results since we do not have any reference as ground truth of the real super-resolved images, we only present the visual results. A detailed difference between a bicubic-trained model and blur-trained model reconstruction from the original can be seen at Figure 6.7.

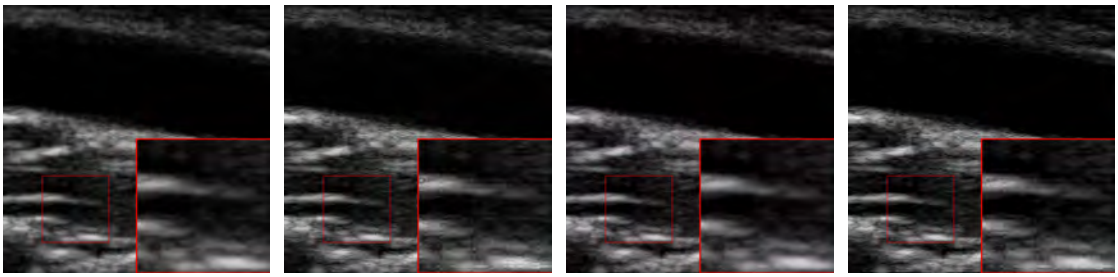
6.5 Discussion

These results show the following points:

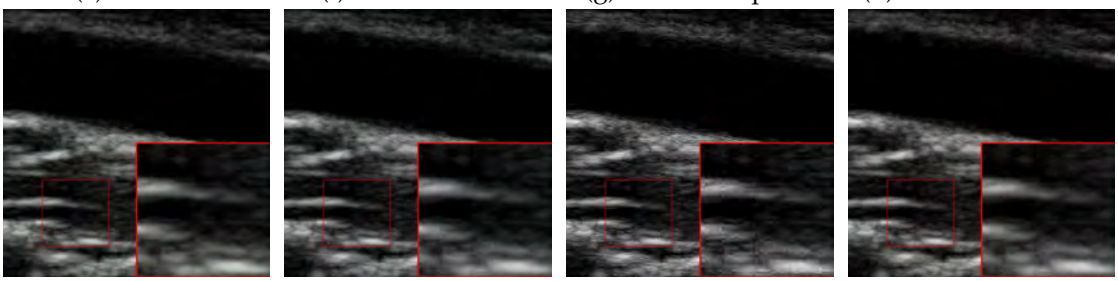
- Models tend to be decisive in terms of metrics (e.g., in all conditions, HAT worked better than DeblurGANv2).
- HAT and HAT-NAF performed better than PRLMU on the CCABlur dataset.
- When comparing the CCA-Blur trained models, HAT and HATNAF worked better than PRLMU in the BUSI-Blur Dataset.
- If blur degradation exists on the dataset tested, blur-trained models report better results; the same applies the other way around.
- Models tend to work better with their datasets (blur-trained model with blur dataset, bicubic-trained model with bicubic dataset).
- Even though blur-fed models tend to create sharper and visually pleasing results, such as shown in Figure 6.7, these may lead to artifacts that need to be checked with the expert/clinician thoroughly.
- Generally, if the dataset trained and tested are the same, it performs better than the counterpart (e.g., CCA-trained models tend to work better on CCA).



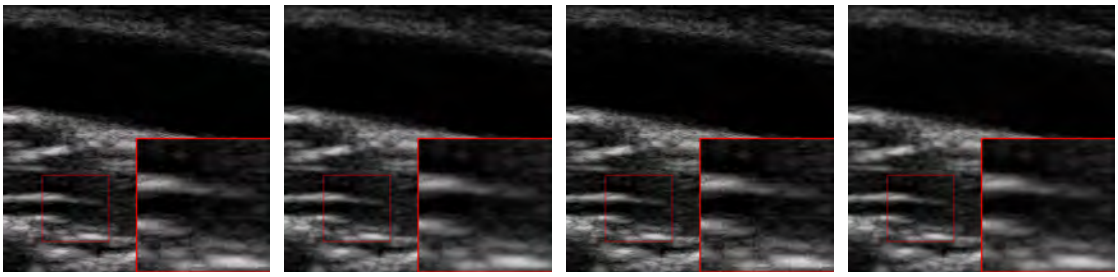
(a) EDSR (b) EDSR - Blur (c) ESRGAN (d) ESRGAN - Blur



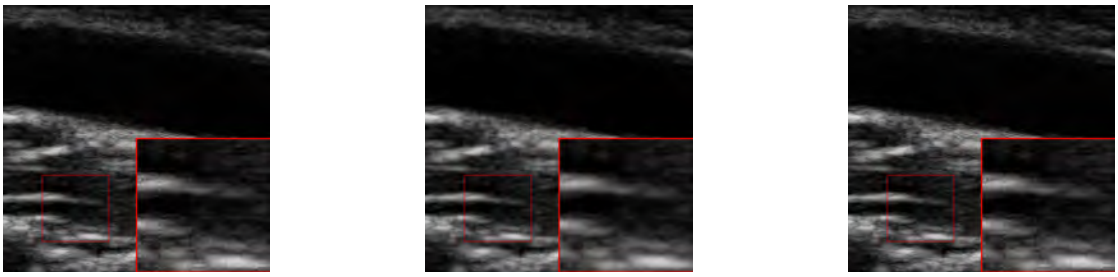
(e) SwinIR (f) SwinIR - Blur (g) DGv2-Inception (h) DGv2-Inc. - Blur



(i) DGv2-Inc. - Aug (j) DGv2-Swinv2 (k) DGv2-Sv2 - Blur (l) DGv2-Sv2 - Aug

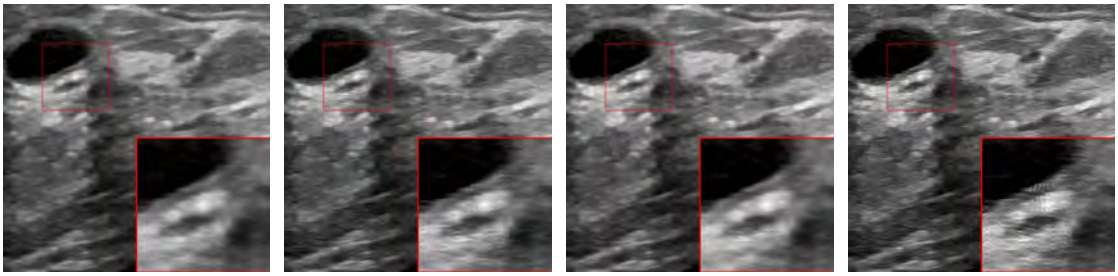


(m) PRLMU - CCABl. (n) NAFNet (o) NAFNet - Blur (p) HAT

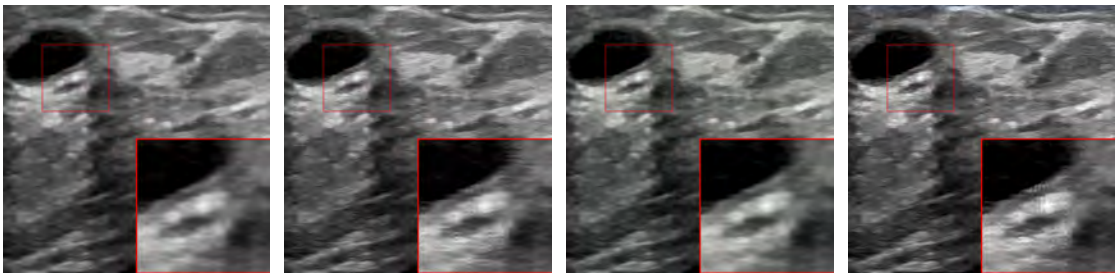


(q) HAT - Blur (r) HATNAF (s) HATNAF - Blur

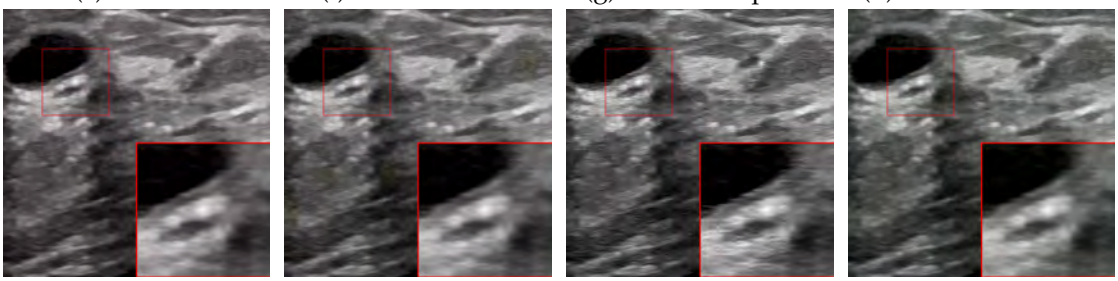
Figure 6.5: CCA results without ground truth



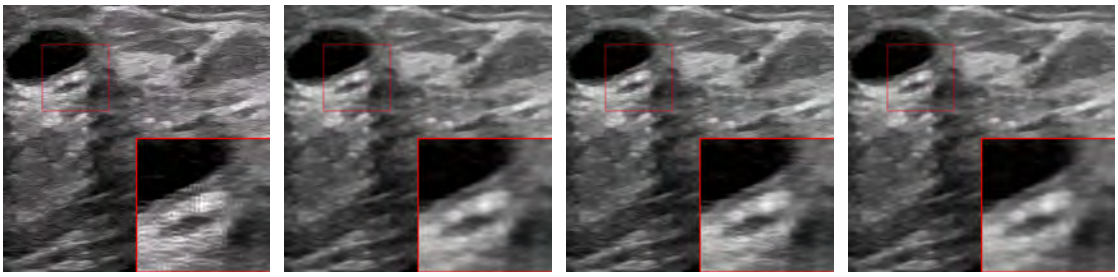
(a) EDSR (b) EDSR - Blur (c) ESRGAN (d) ESRGAN - Blur



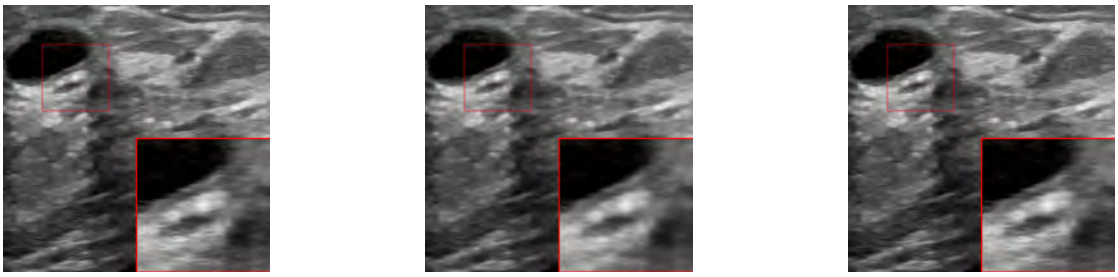
(e) SwinIR (f) SwinIR - Blur (g) DGv2-Inception (h) DGv2-Inc. - Blur



(i) DGv2-Inc. - Aug (j) DGv2-Sv2 (k) DGv2-Sv2 - Blur (l) DGv2-Sv2 - Aug

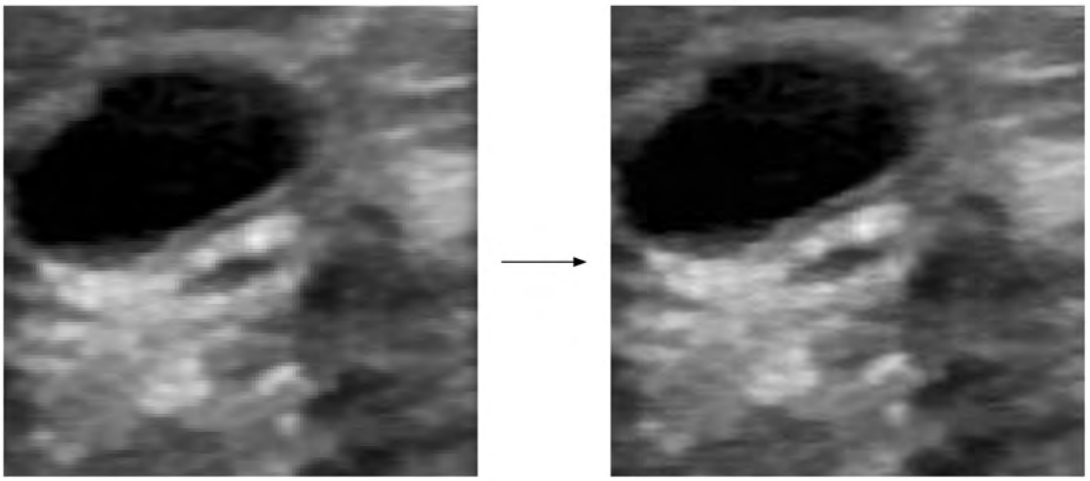


(m) PRLMU - Blur (n) NAFNet (o) NAFNet - Blur (p) HAT

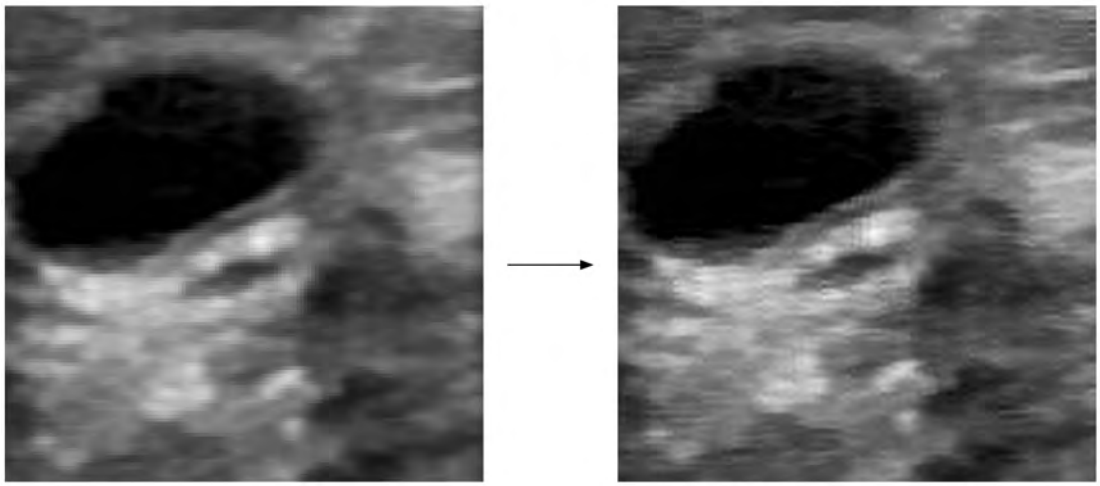


(q) HAT - Blur (r) HATNAF (s) HATNAF - Blur

Figure 6.6: BUSI results without ground truth



(a) Original Image vs. HAT model SR



(b) Original Image vs. HAT-Blur model SR

Figure 6.7: Achieving real SR in the details of the images with different types of models. On the left part, the original image is zoomed via Apple Preview. The right part is the rendering of that area via SR models.

7 Conclusion

By looking at these results, we can conclude some key points. First, by applying SOTA methods in natural images such as HAT, NAFNet, our HAT-NAF mixture, or even SwinIR to the ultrasound images, we achieved comparably better or similar results regarding PSNR and SSIM. Using a deblurring network to realize the noise/blur patterns to fine-tune the image, our proposal also works as intended since it performs enough, especially with the NAFNet architecture.

Despite the claims of SRMD and PRLMU, we have seen that in the datasets of bicubic degradation, the models trained for bicubic degradation and for the datasets of blur degradation, the models trained for blur degradation worked better in terms of metrics, even though blur degradation models produced subjectively visually more pleasing results, due to the sharpening effect, which can turn into an issue that needs to be checked by experts since it could cause artifacts. Therefore, if SR models are trained with the expected degradation model, they tend to perform better instead of degradation-aware degradation optimizations. These results lead to the following question, "how a real super-resolved image be retrieved"? Different solutions can be given, especially with the help of raw signal processing; better resolutions in terms of different types of resolutions, such as axial and lateral resolutions in the spatial domain, can be achieved. For example, by using the same phantom, the two different ultrasound devices can take an ultrasound B-mode image, and the style can be transferred from high quality one to low quality one, or by applying the physics of ultrasound, the same strategy can be done in one ultrasound device, such as using beam-forming techniques can be produced, then by utilizing these deep learning models, new SR images by only using new LR images can be reconstructed. Also, if enough dataset and computational power exist, diffusion-based denoising to synthesize LR images can be done. Another future work might be discovering a degradation learning from ultrasound frequency compounding. Therefore, learning-based SR models can realize and reconstruct better super-resolved images.

Finally, if no external super-resolution solutions exist to transfer the degradation style, by depending on the simulation of SR by having an LR-HR relationship for the taken images, such as in this thesis, a real super-resolved image can be generated. Since no ground truth exists for upscaled images, it's hard to say what is expected. However, models fed with blur-degraded images tend to reconstruct sharpened images with some visual artifacts, such as rings around issues discussed in Figure 2.14. In contrast, models fed with bicubic degraded images tend to reconstruct natural looking

ultrasound (US) images compared to those fed with blur-degraded models, but with a trade-off of more blurry images. Either way, these images will look better than just bicubic upscaled images. When used with a supervision of an expert, we hope these images will be beneficial for ultrasound imaging.

Abbreviations

GAN generative adversarial network

SR super resolution

CNN convolutional neural network

HR high resolution

LR low resolution

PSNR peak signal-to-noise ratio

SSIM structural similarity

PSF point spread function

A-mode amplitude-mode

B-mode brightness-mode

M-mode motion-mode

NLP natural language processing

US ultrasound

MSE mean squared error

EDSR Enhanced Deep Residual Networks for Single Image Super-Resolution

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

DECUSR Deep Convolutional Neural Network for Ultrasound Super Resolution

HAT Hybrid Attention Transformer

Swin Shifted Window Transformer

SwinIR Image Restoration Using Swin Transformer

PRLMU Progressive Residual Learning with Memory Upgrade for Ultrasound Image Blind Super-Resolution

SRMD Super-Resolution Network for Multiple Degradations

IKC Iterative Kernel Correction for Ultrasound Image Super-Resolution

CCA-US common cartoid artery ultrasound

BUSI Breast Ultrasound Images Dataset

FPN Feature Pyramid Network

NN nearest neighbor

ZSSR Zero-Shot Super-Resolution using Deep Internal Learning

USSSCSR Perception Consistency Ultrasound Images Super-Resolution via Self-Supervised CycleGAN

ICAB Improved Channel Attention Block

NAFNet Nonlinear Activation Free Network

SOTA state-of-the-art

CA Channel Attention Module

GELU Gaussian Error Linear Unit

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

ESRGAN Enhanced Super-Resolution Generative Adversarial Networks

SRMD Super-Resolution Network for Multiple Degradations

EDSR Enhanced Deep Residual Networks for Single Image Super-Resolution

SCA Simplified Channel Attention

RRDB Residual in Residual Dense Block

List of Figures

List of Tables

Bibliography

[Al-+20] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy. "Dataset of breast ultrasound images." In: *Data in Brief* 28 (2020), p. 104863. DOI: 10.1016/j.dib.2019.104863.

[Che+22a] L. Chen, X. Chu, X. Zhang, and J. Sun. *Simple Baselines for Image Restoration*. 2022. DOI: 10.48550/ARXIV.2204.04676.

[Che+22b] X. Chen, X. Wang, J. Zhou, and C. Dong. *Activating More Pixels in Image Super-Resolution Transformer*. 2022. DOI: 10.48550/ARXIV.2205.04437.

[Chu+22] X. Chu, L. Chen, C. Chen, and X. Lu. *Improving Image Restoration by Revisiting Global Information Aggregation*. 2022. arXiv: 2112.04491 [cs.CV].

[CUH15] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*. 2015. DOI: 10.48550/ARXIV.1511.07289.

[Dau+16] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. *Language Modeling with Gated Convolutional Networks*. 2016. DOI: 10.48550/ARXIV.1612.08083.

[Dos+20] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2020. DOI: 10.48550/ARXIV.2010.11929.

[GAS20] T. Ganokratanaa, S. Aramvith, and N. Sebe. "Unsupervised Anomaly Detection and Localization Based on Deep Spatiotemporal Translation Network." In: *IEEE Access* PP (Mar. 2020), pp. 1–1. DOI: 10.1109/ACCESS.2020.2979869.

[Gu+19] J. Gu, H. Lu, W. Zuo, and C. Dong. "Blind super-resolution with iterative kernel correction." In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[Gul+17] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. "Improved Training of Wasserstein GANs." In: *CoRR* abs/1704.00028 (2017). arXiv: 1704.00028.

[He+15] K. He, X. Zhang, S. Ren, and J. Sun. *Deep Residual Learning for Image Recognition*. 2015. DOI: 10.48550/ARXIV.1512.03385.

[HG16] D. Hendrycks and K. Gimpel. *Gaussian Error Linear Units (GELUs)*. 2016. DOI: 10.48550/ARXIV.1606.08415.

[Hu+17] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu. *Squeeze-and-Excitation Networks*. 2017. DOI: 10.48550/ARXIV.1709.01507.

[HZ10] A. Horé and D. Ziou. "Image Quality Metrics: PSNR vs. SSIM." In: *2010 20th International Conference on Pattern Recognition*. Aug. 2010, pp. 2366–2369. DOI: 10.1109/ICPR.2010.579.

[Iso+16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. *Image-to-Image Translation with Conditional Adversarial Networks*. 2016. DOI: 10.48550/ARXIV.1611.07004.

[JAF16] J. Johnson, A. Alahi, and L. Fei-Fei. *Perceptual Losses for Real-Time Style Transfer and Super-Resolution*. 2016. DOI: 10.48550/ARXIV.1603.08155.

[Jol18] A. Jolicoeur-Martineau. *The relativistic discriminator: a key element missing from standard GAN*. 2018. DOI: 10.48550/ARXIV.1807.00734.

[KB14] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2014. DOI: 10.48550/ARXIV.1412.6980.

[Kup+18] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas. *DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks*. 2018. arXiv: 1711.07064 [cs.CV].

[Kup+19] O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang. "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better." In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 8878–8887.

[Led+16] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*. 2016. DOI: 10.48550/ARXIV.1609.04802.

[Lia+21] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte. *SwinIR: Image Restoration Using Swin Transformer*. 2021. DOI: 10.48550/ARXIV.2108.10257.

[Lim+17] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. *Enhanced Deep Residual Networks for Single Image Super-Resolution*. 2017. arXiv: 1707.02921 [cs.CV].

[Liu+21a] H. Liu, J. Liu, S. Hou, T. Tao, and J. Han. "Perception consistency ultrasound image super-resolution via self-supervised CycleGAN." In: *Neural Computing and Applications* (2021). DOI: 10.1007/s00521-020-05687-9.

[Liu+21b] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo. "Swin Transformer V2: Scaling Up Capacity and Resolution." In: (2021). DOI: 10.48550/ARXIV.2111.09883.

[Liu+21c] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. 2021. DOI: 10.48550/ARXIV.2103.14030.

[Liu+22] H. Liu, J. Liu, F. Chen, and C. Shan. "Progressive Residual Learning With Memory Upgrade for Ultrasound Image Blind Super-Resolution." In: *IEEE Journal of Biomedical and Health Informatics* 26.9 (Sept. 2022), pp. 4390–4401. ISSN: 2168-2208. DOI: 10.1109/JBHI.2022.3142076.

[LKC16] W. Lotter, G. Kreiman, and D. Cox. *Deep Predictive Coding Networks for Video Prediction and Unsupervised Learning*. 2016. DOI: 10.48550/ARXIV.1605.08104.

[Mao+16] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. *Least Squares Generative Adversarial Networks*. 2016. DOI: 10.48550/ARXIV.1611.04076.

[RFB15] O. Ronneberger, P. Fischer, and T. Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. DOI: 10.48550/ARXIV.1505.04597.

[RL17] C. M. Rumack and D. Levine. *Diagnostic ultrasound*. Elsevier Health Sciences, 2017.

[San+18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." In: (2018). DOI: 10.48550/ARXIV.1801.04381.

[SCI17] A. Shocher, N. Cohen, and M. Irani. *"Zero-Shot" Super-Resolution using Deep Internal Learning*. 2017. DOI: 10.48550/ARXIV.1712.06087.

[SZ14] K. Simonyan and A. Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2014. DOI: 10.48550/ARXIV.1409.1556.

[Sze+16] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*. 2016. DOI: 10.48550/ARXIV.1602.07261.

[TB20] H. Temiz and H. S. Bilge. "Super Resolution of B-Mode Ultrasound Images With Deep Learning." In: *IEEE Access* 8 (2020), pp. 78808–78820. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2990344.

[Vas+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention Is All You Need*. 2017. DOI: 10.48550/ARXIV.1706.03762.

[Wan+18] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang. *ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks*. 2018. arXiv: 1809.00219 [cs.CV].

[Wan+22] X. Wang, L. Xie, K. Yu, K. C. Chan, C. C. Loy, and C. Dong. *BasicSR: Open Source Image and Video Restoration Toolbox*. https://github.com/XPixelGroup/BasicSR. 2022.

[Zam+21] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang. *Restormer: Efficient Transformer for High-Resolution Image Restoration*. 2021. DOI: 10.48550/ARXIV.2111.09881.

[Zha+15] N. Zhao, A. Basarab, D. Kouame, and J.-Y. Tourneret. "Joint Bayesian deconvolution and pointspread function estimation for ultrasound imaging." In: *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*. Apr. 2015, pp. 235–238. DOI: 10.1109/ISBI.2015.7163857.

[Zhu+17] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. 2017. DOI: 10.48550/ARXIV.1703.10593.

[Zuk+13] M. Zukal, R. Beneš, P. Číka, and K. Říha. *Ultrasound image database*. Nov. 2013.

[ZZZ17] K. Zhang, W. Zuo, and L. Zhang. *Learning a Single Convolutional Super-Resolution Network for Multiple Degradations*. 2017. DOI: 10.48550/ARXIV.1712.06116.