AI Medicine
https://www.sciltp.com/journals/aim

Article

Ultrasonic Image's Annotation Removal: A Self-Supervised Noise2Noise Approach

Yuanheng Zhang [1], Nan Jiang [2], Zhaoheng Xie [3], Junying Cao [2,*] and Yueyang Teng [1,*]

[1] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110016, China
[2] The Department of Ultrasound, General Hospital of Northern Theater Command, Shenyang 110169, China
[3] The Institute of Medical Technology, Peking University, Beijing 100191, China
* Correspondence: shenzongchaosheng@163.com (J.C.); tengyy@bmie.neu.edu.cn (Y.T.)

How To Cite: Zhang, Y., Jiang, N., Xie, Z., Cao, J.; Teng, Y. Ultrasonic Image's Annotation Removal: A Self-Supervised Noise2Noise Approach. *AI Medicine* **2024**, *1*(1), 4. https://doi.org/10.53941/aim.2024.100004.

Received: 11 March 2024
Revised: 25 May 2024
Accepted: 28 May 2024
Published: 17 July 2024

Abstract: Accurately annotated ultrasonic images are vital components of a high-quality medical report. Hospitals often have strict guidelines on the types of annotations that should appear on imaging results. However, manually inspecting these images can be a cumbersome task. While a neural network could potentially automate the process, training such a model typically requires a dataset of paired input and target images, which in turn involves significant human labor. This study introduces an automated approach for detecting annotations in images. This is achieved by treating the annotations as noise, creating a self-supervised pretext task and using a model trained under the Noise2Noise scheme to restore the image to a clean state. We tested a variety of model structures on the denoising task against different types of annotation, including body marker annotation, radial line annotation, etc. Our results demonstrate that most models trained under the Noise2Noise scheme outperformed their counterparts trained with noisy-clean data pairs. The costumed U-Net yielded the most optimal outcome on the body marker annotation dataset, with high scores on segmentation precision and reconstruction similarity. Our approach streamlines the laborious task of manually quality-controlling ultrasound scans, with minimal human labor involved, making the quality control process efficient and scalable.

Keywords: image restoration; Noise2Noise; segmentation; U-Net; ultrasonic

1. Introduction

Annotations, typically comprised of various labels and marks, are commonly utilized to record critical information from an ultrasonic exam, including the precise location of potential lesions or suspicious findings, on archived results. Such annotations prove beneficial in aiding physicians in interpreting the exam results, particularly when surrounding structures do not provide any indication of the anatomic location of the image. Additionally, hospitals often mandate the inclusion of annotations, especially in cases involving inter-hospital patient transfers [1]. If the report lacks comprehensive annotations, patients are usually required to undergo an equivalent radiography exam at the facility of transfer.

Commonly employed types of annotations include body marker annotation [2], radial line annotation, and vascular flow annotation. The presence of these annotations serves as evidence for the standardization of the diagnostic process. Annotations not only document the reasoning behind the diagnostic assessment but also facilitate comparison between pre- and post-treatment imaging findings to gain further insights into the patient's condition.

However, the utilization of annotations during ultrasound exams may vary depending on the proficiency of the sonographer performing the procedure. Ultrasound being a live examination makes it hard to implement

additional reviews, thereby relying solely on the expertise of the operator to determine the presence of annotations. Furthermore, the need for repetitive manual verification increases the likelihood of forgetting the task, particularly during busy schedules at hospitals. As such, it is possible for the absence of annotations to occur.

Given the strict regulations and obvious beneficiation surrounding the need for annotations in medical imaging, sonographers need to manually validate that the stored data satisfies these requirements to ensure that diagnoses meet the standard continuously. However, this is a cognitively demanding undertaking as it entails the fulfillment of diverse annotation obligations tailored to specific image outcomes. In addition, dealing with archived files manually is a cumbersome task as most medical data management systems do not consider this necessary and have no relevant feature implemented.

The utilization of neural networks for the automatic assessment of whether the stored data meets particular criteria is a logical approach. To address the current issue, several approaches can be adopted using different types of deep learning models. The first approach would involve treating the task as a semantic segmentation problem, where the goal is to classify each pixel in the image into one of several predefined categories. Alternatively, the task could be framed as an instance segmentation problem, where the aim is to identify and label individual objects within the scene. To accomplish these goals, attention-based models such as the Pyramid Attention Network [3] or the Reverse Attention Network [4] could be employed. Alternatively, generative models like variants of Generative Adversarial Networks (GANs) [5] are also viable. Regardless of the method, once segmentation is completed successfully, the results can be utilized to ascertain the presence or absence of an annotation. (To illustrate, the determination can be achieved by examining the number of white pixels that remain following the application of a filter designed to eliminate noise on the segmentation result.)

This task could also be viewed as an object recognition challenge, and for this purpose, models such as Single Shot MultiBox Detector (SSD) [6] or You Only Look Once (YOLO) [7] could be utilized to obtain the four coordinates of the bonding box of a detected object, which will serve as demonstrative evidence of the necessary annotations.

To train a model using deep learning, it is important to have a suitable training dataset that includes paired input and output data, regardless of the specific task being performed. However, building an appropriate training dataset is a challenging task due to the absence of high-quality data such as segmentation masks, object coordinates and clean targets. Acquiring such data requires a considerable amount of manual effort.

Addressing the challenge of limited labeled data for annotation recognition, this study proposes a self-supervised Noise2Noise approach. The Noise2Noise method stands as a novel training paradigm that departs from the conventional Noise2Clean approach. Unlike Noise2Clean, which necessitates paired noisy-clean image datasets, Noise2Noise leverages innovative mathematical principles to train a denoising model solely on noisy data. This eliminates the requirement for a large and often impractical collection of clean images.

Building upon the Noise2Noise framework, we propose a novel self-supervised strategy, where common annotations are treated as noise and randomly superimposed, in a repetitive manner, onto a limited set of unannotated images. This process effectively generates a synthetic training dataset specifically tailored for the Noise2Noise framework. The trained model, equipped to remove noise (in this case, annotations), can then be employed for annotation recognition without requiring clean image counterparts.

We trained multiple network structures such as FCN, U-Net++, MultiResUNet, etc., under both training paradigms to select an ideal one. We noted that the majority of Noise2Noise-based methods surpassed the corresponding Noise2Clean (supervised learning) methods in which the former even received a Sørensen-Dice coefficient (Dice) increase of up to 300%, an Intersection over Union (IoU) increase of up to 384%, and a Peak Signal to Noise Ratio Human Visual System Modified (PSNR HVS M) increase of up to 38% in some cases. Among them, our costumed U-Net achieved the best results, both quantitative and qualitatively.

The remainder of the paper is organized as follows: Section 2 discusses related works. Section 3 outlines our methodology, data sources, dataset-building pipeline, and model structures used in this work. In Section 4, quantitative metric scores and qualitative image results are provided to support our claim regarding the optimal model structure, loss function, and observations on Noise2Noise's effect. Finally, Section 5 concludes the paper.

2. Related Work

2.1. Self-Supervised Learning

Self-supervised learning is a way of training deep-learning models without human guidance or explicit instructions. Unlike supervised learning which uses labeled examples, self-supervised models learn from unlabeled data by identifying patterns and relationships on their own. It uses the structure of images (e.g., edges, shapes) to

teach the deep-learning model how to identify important parts of an image automatically, rather than having to be explicitly told what to look for. This is particularly helpful considering the abundance of unlabeled data that exists today and the amount of work required to create a properly constructed dataset. To create a robust, large model, self-supervised learning is an essential tool.

The general process of self-supervised learning involves first creating a pretext task for the model to solve. By completing this task, the model can gain an understanding of the structural information embedded within the data. This understanding can then be transferred to downstream tasks using different forms of transfer learning.

Examples of pretext tasks include rotating an image for the model to predict the degree of rotation, reconstructing images from an altered view, or reconstructing images from a corrupted version of the original data.

In this work, we developed a pretext task where we asked the model to generate another noisy image from the noisy input while keeping the same original clean image beneath it. Specifically, we manually extracted several common annotations from stored data and randomly superimposed them on a small set of unannotated images to create a large dataset. The idea behind this approach was to train the model to recognize the crucial features of the original so that it could distinguish between noise and clean images.

2.2. Noise2Noise Training Scheme

Noise2Noise is originally proposed in [8] as a novel statistical reasoning for the task of image denoising. It is shown that, under certain key constraints, it is possible to train a denoising model using only corrupted images. The constraints are: the distribution of the added noise must have a mean of zero and no correlation with the desired clean image, and the correlation between the noise in the input image and the target image should be close to zero [9].

By utilizing deep learning, a denoising task can be transformed into a regression problem, where a neural network is used to learn the mapping between corrupted samples $\hat{x}_i$ and clean samples y_i by minimizing the empirical risk [8].

In [8], inspecting the form of a typical training process shows that training a neural network is a generalization of a point estimating problem. We can see that it is essentially solving the point estimating problem for each separate input. This means that by finding the optimal parameters, the trained neural network will output the expectation or median of all possible mapping for input x. This property often leads to unwanted fuzziness in many deep-learning applications. However, in a denoising scenario, when the noise satisfies the above constraints and exists in both the model input and training target, the task of empirical risk minimization, given infinite data,

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(\hat{x}_i), \hat{y}_i) \tag{1}$$

is equivalent to the original regression problem

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(\hat{x}_i), y_i) \tag{2}$$

where $f_\theta(x)$ is the model parameterized by θ, L is the loss function, $\hat{x}_i, \hat{y}_i$ are samples drawn from a noisy distribution and y_i representing clean samples.

The idea of using self-supervised learning in conjunction with Noise2Noise training scheme aligns well with our goal of obtaining a clean image. With a clean image, we can easily produce a segmentation map for various kinds of annotations, facilitating the models to recognize and categorize them accurately.

3. Methodology

Building on the aforementioned theories, we address the challenge of limited data for segmentation by treating the desired object (annotations, in this paper) as noise. After we create a Noise2Noise dataset, we train a denoising model to remove this object. The resulting denoised image, when subtracted from the original input, provides a segmentation mask. This mask allows us to determine the presence of the target object with a simple score based on the number of white pixels.

To be more specific, initially, our data includes collections of data that may or may not have specific annotations. We manually examined and filtered the data to create a clean dataset for each annotation. Next, we studied the individual components of different annotations and identified a general pattern for each one. Using this pattern, we generated large datasets containing noisy data and trained a denoising model using the Noise2Noise approach, and designed a pretext task with this dataset. Finally, we trained various model structures using both the

Noise2Noise and conventional Noise2Clean techniques to obtain denoising models for performance comparison (based on the denoised result and segmentation mask).

3.1. Dataset

To manually synthesize a self-supervised Noise2Noise dataset, which our training requires, it is essential to know the scheme of the different annotations and to construct a dataset according to it.

Our original data consists mainly of ultrasonic images provided by the General Hospital of Northern Theater Command. These images were captured using external video capture cards and are in 8-bit sRGB format.

According to the type of noise, we divided these data into six categories:

- Images with body marker annotation
- Images without body marker annotation
- Images with radial line annotation
- Images without radial line annotation
- Images with vascular flow annotation
- Images without vascular flow annotation

Images with certain annotations are considered noisy images in the context of the noise removal task, and corresponding images without these annotations are considered clean. Some typical images with various annotations are provided in Figure 1.

To safeguard the confidentiality of the patient, any personal data displayed in the margin of the image is blurred using pixelization. This same technique is also used to obscure any similar information present in other images.

In essence, a body marker annotation is a marker selected from a fixed set of icons that indicates different regions of the human body and its current orientation. It is typically located at the edge of the ultrasonic image area and is labeled by the sonographer. On some ultrasound machines, the body marker annotation has a fixed position.

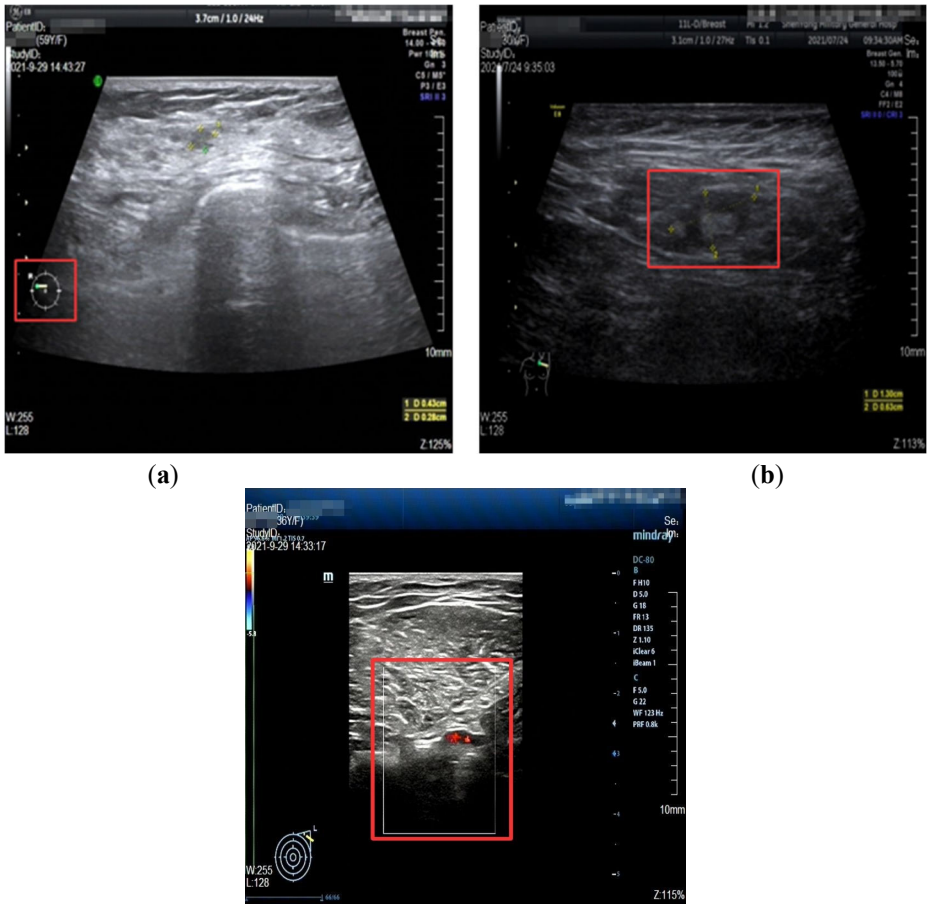


Figure 1. Images with various annotations: (**a**) body marker annotation; (**b**) radical line annotation; (**c**) vascular flow annotation.

However, from a statistical and training perspective, each real instance can be viewed as an image sample from a conditional distribution where the condition is the body marker annotation's location. By randomly placing body marker annotation at any position within the image, we draw samples from a distribution without the aforementioned condition. By learning to denoise samples from the unconditioned distribution, the model can effectively denoise samples from conditional distribution as well.

While we introduced randomness to annotation shapes, we did not completely randomize their placement. After analyzing existing data, we observed that body marker annotations rarely appear in the image center. So, we limited the program to placing annotations only within a 20% border around the image edges.

Other commonly used annotations that we introduced later comply with the same reasoning.

The radial line annotation indicates pairs of connected cross markers. They are usually placed at the edge of the lesion area, with its placement determined by the size of the lesion. One to three pairs of cross markers may be present in an image, corresponding to the three axes of 3D space, but typically there are only two pairs.

The vascular flow annotation is not an additional labeling feature meant to simplify identification. Rather, it serves as a bounding box that identifies the specific area of the image being examined by the ultrasound flowmeter. However, to keep things simple, we will continue to call it a form of annotation. The presence of this annotation indicates that the relevant examination has been conducted.

To synthesize a Noise2Noise training dataset for the above annotations, we first manually extracted the necessary annotation icons from existing annotated data, and then we randomly overlay different annotations on the clean images we have. To improve the model's ability to handle variations (generalization), we also introduced randomness into the shape of annotations. For instance, the lines connecting markers in vascular flow annotations have a random, constantly changing appearance. This approach accounts for the different annotation styles used by various ultrasound machines. The randomness of the noise overlay allows for the creation of a relatively large dataset.

By constructing training datasets in the above-mentioned process, each noisy image has three corresponding images for different tasks.

- A clean image which the noisy image originated from.
- A different noisy image is created from the same clean image, using a different (in terms of position, form, etc.) noise sampled from the same distribution.
- A binary image recorded the position and form of the noise appended to the clean image.

An instance of the training dataset is presented in Figure 2. Using these images, the same dataset can be used for Noise2Noise training, conventional Noise2Clean training, and normal segmentation training.

Our approach to creating this training dataset can minimize the amount of human labor required. Even with a limited amount of clean data, we can generate a large noisy dataset for training. The flow chart of the above process is also shown in Figure 2.

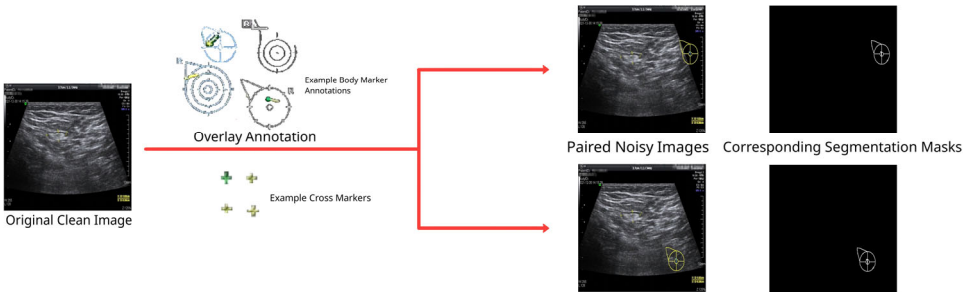


Figure 2. Flow chart of training dataset building.

3.2. Network Structures

In this research, we trained several structures to find the optimal solution and compare the two different training schemes: Noise2Noise and traditional Noise2Clean.

We adopted most of the structures from the traditional image segmentation model. The models we adopted include FCN, DeepLabv3, LinkNet, MANet, U-Net Plus Plus, MultiResUNet and a costumed U-Net.

FCN is one of the models utilizing convolutional networks in semantic segmentation. Long et al. [10,11] use fully-convolutional layers instead of fully-connected layers so that this model is compatible with non-fixed sized input and ouputs.

DeepLabv3 is a subsequent model of the DeepLab model family, developed by Chen et al. [12]. The main feature of this model is the use of dilated convolution, also known as "atrous" convolution. This method is advocated to combat the issue of feature resolution reduction in deep convolutional networks (due to pooling operations and strides in convolution operations) and the difficulties in multi-scale segmentation.

LinkNet is proposed by Chaurasia and Culurciello [13] to address the problem of the long processing time of most segmentation models. By using a skip connection to pass spatial information directly to the corresponding decoder, LinkNet manages to preserve low-level information without additional parameters and re-learning operations.

MANet, or Multi-scale Attention Net, is developed to improve accuracy in semantic segmentation of remote sensing images. By using a novel attention mechanism, treating attention as a kernel function, Li et al. [14,15] reduce the complexity of the dot product attention mechanism to $O(N)$.

U-Net is a well-known encoder-decoder segmentation model. It is originally proposed by Ronneberger et al. [16,17] for segmenting biological microscopy images.

U-Net++ is a variant of U-Net proposed by Zhou et al. [18]. In their work, they proposed a novel skip connection block in which a dense convolution block is used to process the input from the encoder feature map so that the semantic level of the input is closer to the corresponding decoder feature map.

MultiResUNet is another modern variant of U-Net proposed by Ibtehaz and Rahman [19] as a potential successor. They used an Inception-like layer to replace the consecutive convolution layers after each pooling and transpose-convolution layers, to percept objects at different scales. They adopted a chain of convolution layers with residual connections instead of plain skip connection to process the feature map inputs before concatenating them to decoder feature maps.

In our work, since the vanilla U-Net does not match the spatial resolution of our dataset, we used a costumed U-Net similar to [8] in all of our tests. Our architecture utilizes convolutional layers with strategically chosen stride and padding values to maintain consistent spatial dimensions between the network's input and output. Within the costumed U-Net implementation employed in this work, the encoder stage leverages 3x3 convolutional kernels with a stride of 2 and padding of 1. This configuration progressively increases the feature map dimensionality (from 3 to 38, 96, and finally 144) while downsampling the spatial resolution. The corresponding decoder stage mirrors these convolutional layers to achieve dimensionality reduction (from 144 to 96, 38, and finally 3). To achieve upsampling within the decoder, transposed convolutional layers are employed with parameters identical to their corresponding counterparts in the encoder. ReLU activation is utilized as the non-linearity after each convolutional or transposed convolutional layer, except for the final output layer which employs LeakyReLU activation. This specific configuration ensures that the processed data retains its original spatial dimensions throughout the costumed U-Net architecture. The detailed structure is presented in Table 1.

Table 1. Detailed structure of our costumed U-Net.

Layer Name	N_out (Channels)	Function
Input	3	-
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
MaxPool2d	48	Max Pooling (2 × 2 kernel, Stride 2 × 2, Padding 0 × 0)
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
MaxPool2d	48	Max Pooling (2 × 2 kernel, Stride 2 × 2, Padding 0 × 0)
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
ConvTranspose2d	48	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
Concat	96	Concatenate Feature Maps
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
ConvTranspose2d	96	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)

Concat	144	Concatenate Feature Maps
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
ConvTranspose2d	96	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
Concat	99	Concatenate Feature Maps
Conv2d	64	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	64	ReLU activation
Conv2d	32	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	32	ReLU activation
ConvTranspose2d	3	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
LeakyReLU	3	LeakyReLU activation

4. Results

In this section, we provide performance evaluations and comparative studies.

4.1. Evaluation

We evaluate the model's performance based on segmentation precision and reconstruction similarity.

4.1.1. Segmentation Precision

In terms of noise reduction precision, for a typical segmentation model, we can use the output to compare it with a binary image known as the truth mask to compute a score based on the number of pixels that get classified into the right categories. For a restoration model like ours, we subtract the model output from the model input to compute the binary segmentation result. We compare the results with the segmentation truth mask to compute the Dice, IoU, and Pixel Accuracy (PA).

4.1.2. Reconstruction Similarity

For assessing reconstruction similarity, we use two metrics: Structural Similarity Index Measure (SSIM) and PSNR_HVS_M. SSIM is a commonly used measure of image similarity. The PSNR metric known as PSNR_HVS_M [20] is considered to be a more accurate representation of image quality, which takes into consideration the Contrast Sensitivity Function (CSF) and the between-coefficient contrast masking of Discrete Cosine Transform (DCT) basis functions.

4.2. Training

The neural networks discussed in the previous section were trained using PyTorch 1.10.1. RMSprop [21], a variant of stochastic gradient descent that divides gradients by an average of their recent magnitude, was used as the optimizer with a learning rate of 0.00001, momentum of 0.9, weight decay of $1e^{-8}$, and default values [22] for other parameters.

Three datasets were created in the aforementioned process to train various denoising models. For body marker annotation, a dataset of 83,900 pairs of noisy images generated from 4,975 clean images was used. For radial line annotation, 80,000 pairs of noisy images were generated from 3,936 clean images. For vascular flow annotation, 80,000 pairs of noisy images were generated from 250 clean images.

4.3. Optimal Model Structure

To find the most effective combination of network structure and training scheme for the given task, we trained different network structures under the Noise2Noise and Noise2Clean schemes using the body mark annotation dataset. Though utilizing only one type of annotation, this experiment's results could demonstrate the likely most suitable structure for other annotations as well. L_1 loss is used to train these models. The results were compared using segmentation precision and reconstruction similarity, and are presented in Tables 2 and 3.

We observed that Noise2Noise training scheme improves segmentation precision and reconstruction similarity in most cases. The results presented in Tables 2 and 3 indicate that the models trained using the

Noise2Noise scheme generally achieved higher Dice scores, IoU scores, PA scores, and PSNR_HVS_M scores. Specifically, for the costumed U-Net, we observed an increase in the Dice and IoU of 0.151 and 0.155, respectively, and an increase of 11.625 for the PSNR_HVS_M when using linearly normalized input.

According to our hypothesis, the Noise2Noise training process improves the model's ability to understand the features of annotations through solving an "impossible" task of relocating the annotation. This task is essentially a self-supervised pretext training task that helps the model gain a better understanding of the annotations and the spatial structure of the ultrasonic images, thus gaining higher performance. To highlight the advantage of our approach, let's consider the limitations of traditional Noise2Clean denoising. In Noise2Clean, the neural network learns convolutional kernels to remove noise. These kernels may develop a complex mask that essentially averages pixels within an applied area. This approach circumvents the need for the model to learn the specific relationship between the target noise and the underlying clean image. The acquisition of structural features by the model is not guaranteed. During training, the model may converge on an optimal solution that captures these features, leading to successful performance. However, the possibility exists that the model converges at a local optimum, neglecting this information even with abundant training data. Conversely, our Noise2Noise approach with overlaid annotations trains the model to essentially move the annotation (treated as noise) within the image. This process necessitates learning the structural information of both the annotation and the underlying clean background. It's well-established that core self-supervised learning tasks, such as rotation prediction, jigsaw puzzle solving, and missing patch prediction, all hinge on the model's ability to grasp structural information. Our proposed task shares this very property. We posit that this fundamental difference in training objectives is a key factor contributing to the performance improvement observed in our method.

We also noted that the costumed U-Net structure performed the best out of all the structures tested. It achieved the highest Dice, IoU, SSIM, and PSNR_HVS_M scores under both training schemes. The costumed U-Net trained using the Noise2Noise scheme achieved the highest segmentation precision and reconstruction similarity of all models, with a Dice of 0.712, an IoU of 0.596, an SSIM of 0.967, and a PSNR_HVS_M of 41.628.

Our findings suggest that a model's capacity to retain graphical details is a critical performance factor. As shown in Tables 2 and 3, a significant performance gap exists between models employing skip connections (facilitating detail preservation) and those lacking such structures. Interestingly, our results indicate that a concise skip connection pathway is preferable for this task. Models with intricate skip connection architectures (such as U-Net++ and MultiResUNet) exhibited lower performance compared to U-Net's straightforward skip connections. This might be attributed to the desired outcome: preserving most of the input information in the output, only removing annotation in an area of interest. Therefore, simpler skip connection pathways are preferred. Complex architectures introduce additional weights and parameters, potentially hindering the model's ability to faithfully transmit the input information.

Given the above results, we chose the costumed U-Net as the optimal model for later experiments.

Table 2. Segmentation Precision on Body Marker Annotation (Average + Var) N2C stands for Noise2Clean, N2N stands for Noise2Noise SMN indicates the model is trained with data normalized according to standard deviation and mean Models without SMN are trained with linearly normalized data.

Method	Training Mode	Dice	IoU	PA
FCN_101	N2C SMN	0.07 ± 0.003	0.039 ± 0.001	$0.97 \pm 7.2 \times e^{-5}$
FCN_101	N2N SMN	0.07 ± 0.003	0.04 ± 0.001	$0.97 \pm 8 \times e^{-5}$
DeepLab V3	N2C	0.073 ± 0.003	0.039 ± 0.001	0.969 ± 0.005
DeepLab V3	N2N	0.074 ± 0.003	0.04 ± 0.001	0.969 ± 0.005
LinkNet	N2C	0.447 ± 0.105	0.346 ± 0.007	0.976 ± 0.008
LinkNet	N2N	0.343 ± 0.139	0.280 ± 0.106	0.938 ± 0.008
MANet	N2C	0.531 ± 0.113	0.430 ± 0.091	0.943 ± 0.015
MANet	N2N	0.543 ± 0.128	0.451 ± 0.105	0.917 ± 0.024
U-Net++	N2C	0.551 ± 0.08	0.437 ± 0.07	0.983 ± 0.007
U-Net++	N2N	0.613 ± 0.114	0.516 ± 0.09	0.943 ± 0.016
MultiResUNet	N2C SMN	0.416 ± 0.05	0.594 ± 0.04	$0.998 \pm 2.75 \times e^{-6}$
MultiResUNet	N2N SMN	0.661 ± 0.06	0.539 ± 0.06	$0.99 \pm 5 \times e^{-4}$
Costumed U-Net	N2C SMN	0.408 ± 0.05	0.286 ± 0.04	$0.998 \pm 2.54 \times e^{-6}$
Costumed U-Net	N2N SMN	0.676 ± 0.05	0.552 ± 0.05	$0.999 \pm 5 \times e^{-7}$
Costumed U-Net	N2C	0.561 ± 0.077	0.441 ± 0.072	0.990 ± 0.005
Costumed U-Net	N2N	0.712 ± 0.053	0.596 ± 0.058	0.993 ± 0.007

Table 3. Reconstruction Similarity on Body Marker Annotation (Average + Var).

Model	Training Mode	SSIM	PSNR_HVS_M
FCN_101	N2C	0.459 ± 0.001	10.264 ± 1.751
FCN_101	N2N	0.453 ± 0.016	10.181 ± 2.430
DeepLab V3	N2C	0.680 ± 0.004	15.919 ± 2.578
DeepLab V3	N2N	0.678 ± 0.005	15.827 ± 3.282
LinkNet	N2C	0.933 ± 0.000	25.691 ± 6.425
LinkNet	N2N	0.945 ± 0.000	26.307 ± 8.466
MANet	N2C	0.923 ± 0.002	21.920 ± 7.015
MANet	N2N	0.923 ± 0.002	23.027 ± 3.903
U-Net++	N2C	0.923 ± 0.000	21.245 ± 1.846
U-Net++	N2N	0.927 ± 0.000	24.366 ± 7.121
MultiResUNet	N2C SMN	0.856 ± 0.002	23.712 ± 3.936
MultiResUNet	N2N SMN	0.792 ± 0.004	21.256 ± 6.160
Costumed U-Net	N2C SMN	0.833 ± 0.003	11.828 ± 20.299
Costumed U-Net	N2N SMN	0.791 ± 0.004	20.746 ± 10.223
Costumed U-Net	N2C	0.961 ± 0.000	29.976 ± 30.140
Costumed U-Net	N2N	0.967 ± 0.000	41.628 ± 41.775

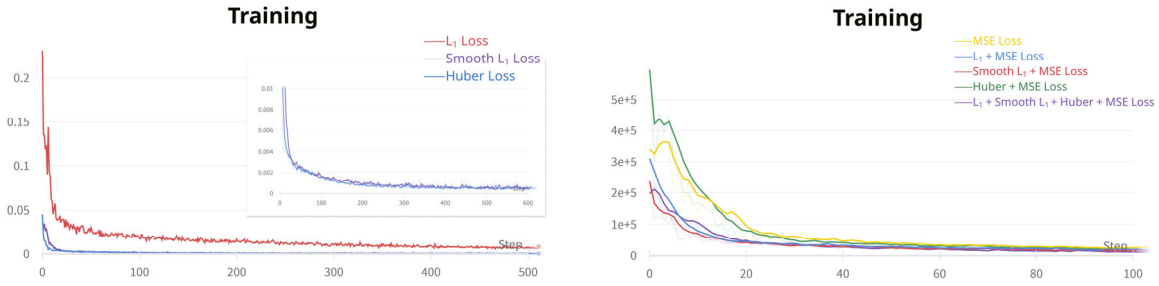
4.4. Optimal Loss Function

To find the optimal loss function, we evaluate the convergence speed of different loss functions. The loss functions we tested include L_1 loss, Huber loss, Smooth L_1 loss, MSE loss and several combinations of aforementioned loss functions. The result is shown in Figure 3.

To better visualize the differences in convergence speed between the losses, we present them in separated subplots. As shown in Figure 3a, the L_1 loss and its variants (Huber loss and Smooth L_1 loss) are displayed on one subplot, while the MSE loss-related losses are presented on another subplot in Figure 3b.

We observed that implementing MSE loss results in faster convergence, allowing the model to reach convergence in under 100 steps, as shown in Figure 3b. Meanwhile, as depicted in Figure 3a, the loss functions based on L_1 loss achieve a much slower convergence after approximately 500 to 600 steps. Although Huber loss and Smooth L_1 loss seem to have a quicker rate of convergence, closer examination in Figure 3a reveals that they both take around 500 steps to converge, which is similar to the standard L_1 loss.

We also noted from Figure 3b that using a combination of MSE loss and different L_1 based losses doesn't significantly affect the rate of convergence, likely because the difference in scale between the MSE loss and L_1 loss and its variants causes MSE loss to remain the primary determinant of convergence speed.



(**a**) (**b**)

Figure 3. Loss functions convergence comparison: (**a**) Loss of L_1 and its variants. (**b**) Loss of MSE loss and other combined losses.

Our study also conducted an evaluation of the costumed U-Net trained using various loss functions. Our findings in Tables 4 and 5 revealed that there was minimal difference between the performances of these models, with the largest discrepancies in Dice, IoU, PA, SSIM and PSNR_HVS_M amounting to 0.023, 0.019, 0.003, 0.011 and 4.031, respectively. These outcomes suggest that the selection of alternative loss functions has little influence on the overall performance of the model. As such, we decided not to employ the MSE loss function in subsequent experiments and instead continued to utilize the L_1 loss.

Table 4. Segmentation Precision on Body Marker Annotation for the Costumed U-Net Trained with Different Loss Functions (Average + Var).

Loss Function	Dice	IoU	PA
L 1	0.712 ± 0.053	0.596 ± 0.058	0.993 ± 0.007
Huber	0.708 ± 0.05	0.592 ± 0.005	0.993 ± 0.005
Smooth L1	0.717 ± 0.05	0.599 ± 0.055	0.993 ± 0.005
L2	0.716 ± 0.053	0.599 ± 0.056	0.993 ± 0.005
L1 + L2	0.712 ± 0.053	0.596 ± 0.057	0.993 ± 0.005
Huber + L2	0.713 ± 0.052	0.596 ± 0.057	0.993 ± 0.005
Smooth L1 + L2	0.692 ± 0.068	0.580 ± 0.066	0.990 ± 0.005
All Loss Sum	0.715 ± 0.052	0.598 ± 0.057	0.993 ± 0.005

Table 5. Reconstruction Similarity on Body Marker Annotation for the Costumed U-Net Trained with Different Loss Functions (Average + Var).

Loss Function	SSIM	PSNR_HVS_M
L 1	0.967 ± 0.000	41.628 ± 41.775
Huber	0.968 ± 0.000	38.110 ± 91.416
Smooth L1	0.967 ± 0.000	41.737 ± 38.719
L2	0.966 ± 0.000	40.982 ± 37.608
L1 + L2	0.966 ± 0.000	38.689 ± 46.355
Huber + L2	0.977 ± 0.000	42.141 ± 53.215
Smooth L1 + L2	0.968 ± 0.000	39.186 ± 47.249
All Loss Sum	0.968 ± 0.000	40.443 ± 57.084

4.5. Noise2Noise with Other Annotations

The improvement observed in the costumed U-Net trained using the Noise2Noise scheme is also apparent in other annotation datasets, as shown in Tables 6–9. In the provided tables, the costumed U-Net has been trained using other two annotation datasets along with two different training schemes. The outcomes show a substantial enhancement in comparison to the Noise2Clean models, as there is approximately a half-unit gain observed in both Dice and IoU metrics, an increase of around 0.01 in SSIM, and a rise of 5 units in PSNR_HVS_M for both types of annotations.

The performance improvement observed in the Noise2Noise model further strengthens our hypothesis. This is because both radial line annotations (cross markers) and vascular flow annotations (boxes) share similarities with other highly prevalent elements in ultrasonic imaging results. Models trained with the traditional Noise2Clean approach struggle to develop kernels that can differentiate these desired annotations from other image information. Conversely, the Noise2Noise model circumvents this limitation.

Table 6. Segmentation Precision on Radial Line Annotation (Average + Var).

Method	Training Mode	Dice	IoU
Costumed U-Net	N2C	0.226 ± 0.013	0.132 ± 0.006
Costumed U-Net	N2N	0.747 ± 0.004	0.639 ± 0.059

Table 7. Reconstruction Similarity on Radial Line Annotation (Average + Var).

Method	Training Mode	SSIM	PSNR_HVS_M
Costumed U-Net	N2C	0.932 ± 0.000	21.660 ± 5.391
Costumed U-Net	N2N	0.942 ± 0.000	26.376 ± 0.681

Table 8. Segmentation Precision on Vascular Flow Annotation (Average + Var).

Method	Training Mode	Dice	IoU	PA
Costumed U-Net	N2C	0.243 ± 0.028	0.149 ± 0.013	0.989 ± 1.115
Costumed U-Net	N2N	0.728 ± 0.031	0.599 ± 0.039	0.998 ± 1.423 × e^{-5}

Table 9. Reconstruction Similarity on Vascular Flow Annotation (Average + Var).

Method	Training Mode	SSIM	PSNR_HVS_M
Costumed U-Net	N2C	0.938 ± 0.000	21.584 ± 5.384
Costumed U-Net	N2N	0.948 ± 4.853 × e−5	26.717 ± 0.511

4.6. Qualitative Results

In this section, we present denoised images from models trained under different schemes to further support our claim.

As can be seen in Figures 4–6, the output from the Noise2Clean model contains obvious artifacts, whereas models trained using the Noise2Noise scheme do not suffer from this problem.

It is also worth noting that in the output images from Noise2Clean models, information in the edge area is compromised. In contrast, the Noise2Noise models preserve this information well. The evidence implies that models trained with the Noise2Noise scheme possess superior capabilities in identifying and distinguishing noise.

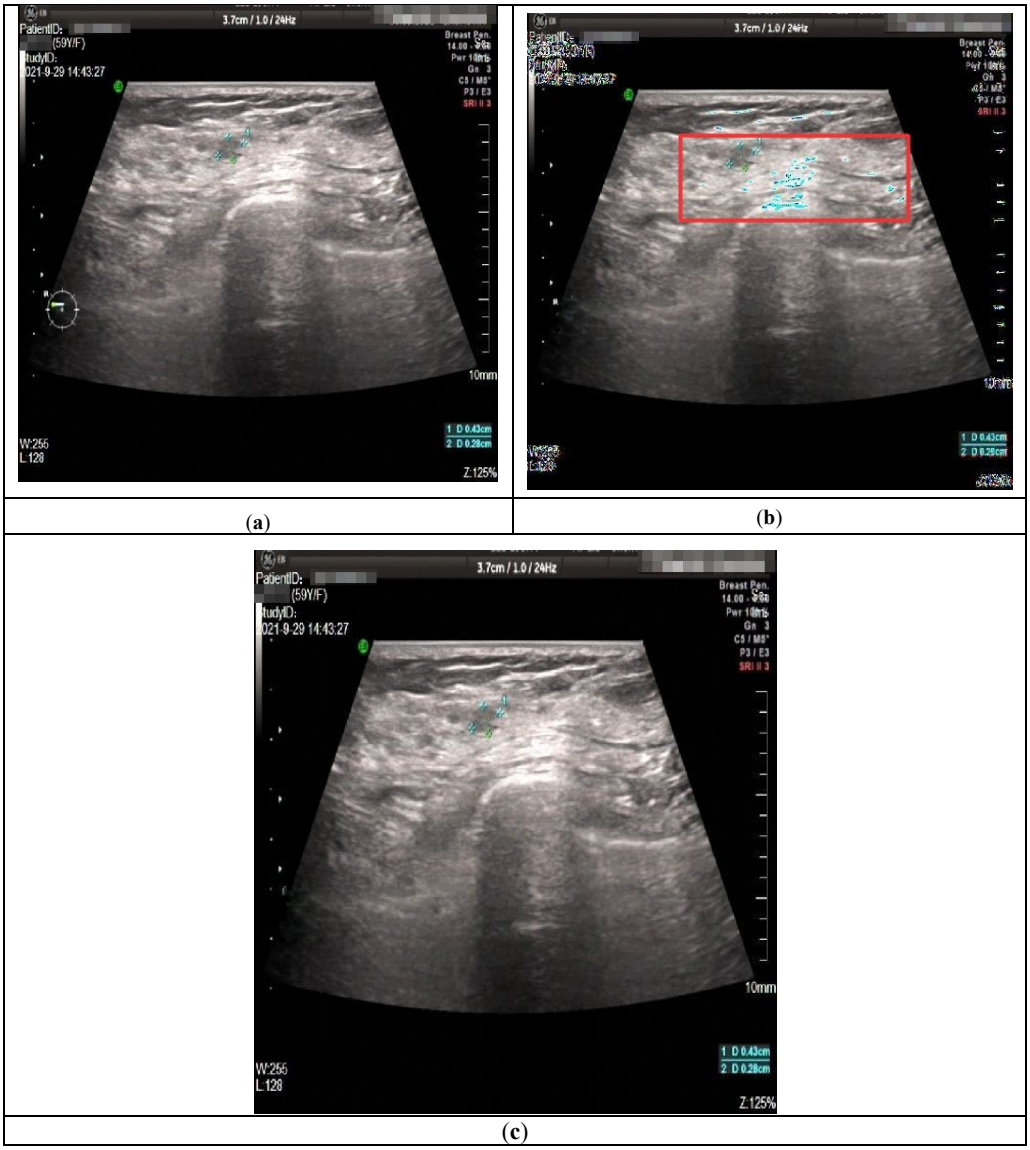
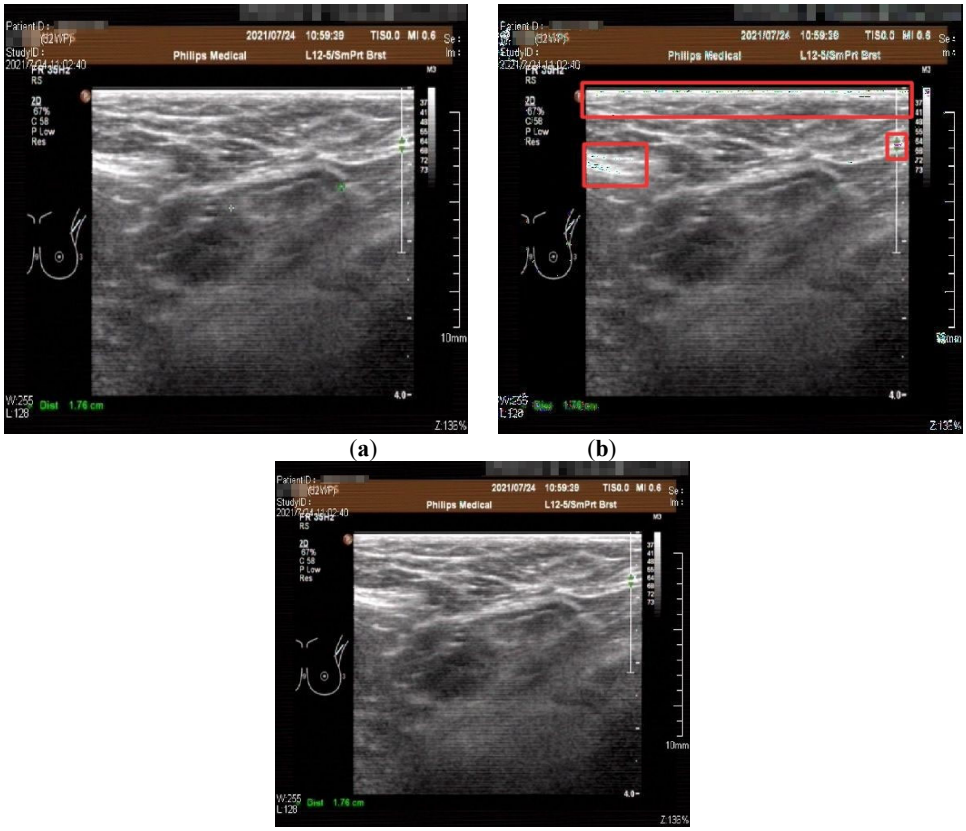


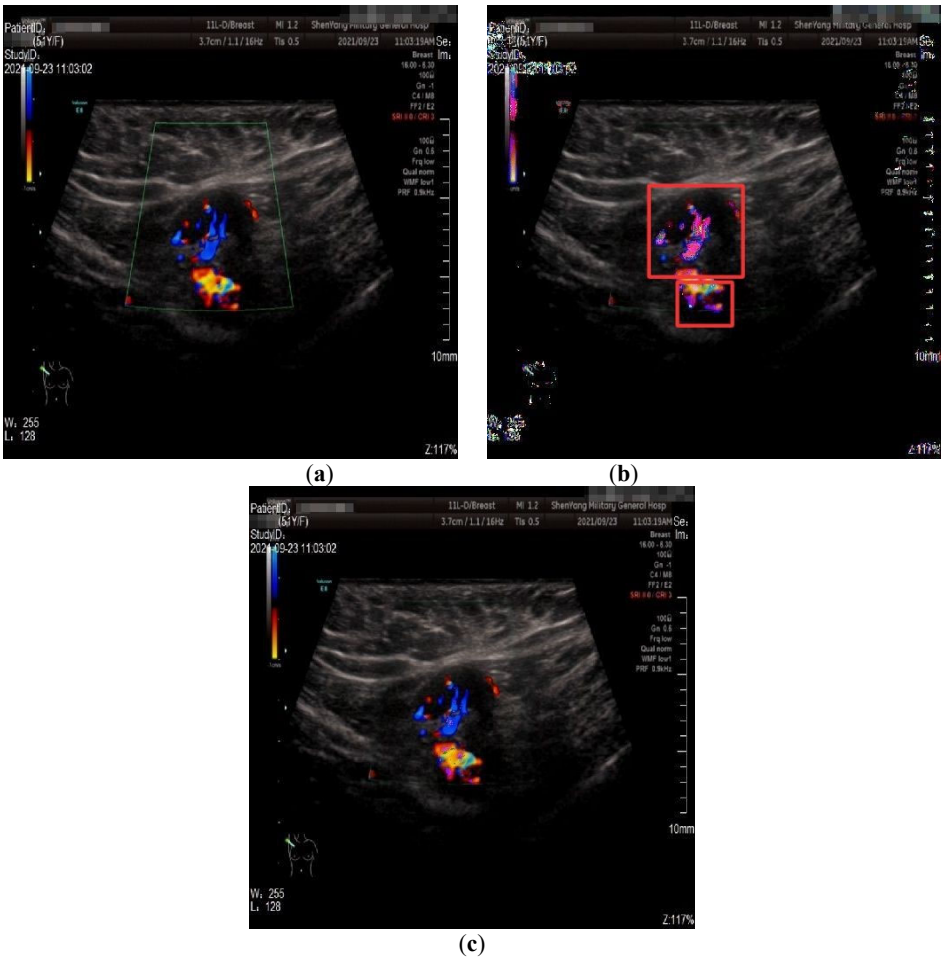
Figure 4. Body marker annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.



(**a**) (**b**)

(**c**)

Figure 5. Radial line annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.



(**a**) (**b**)

(**c**)

Figure 6. Vascular flow annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.

5. Discussion and Conclusions

This study proposed a self-supervised data generation and training approach to build large and diverse datasets starting from a small dataset with only a few clean images. We find that the costumed U-Net trained with the Noise2Noise scheme outperformed other models in terms of segmentation precision and reconstruction similarity in the annotation removal task. The benefits of Noise2Noise training were observed across most model structures tested, and the models trained using this scheme produced fewer artifacts.

Our study has some limitations: Firstly, we used separate parameter sets for the segmentation task of different annotations. However, with the recent advancement of deep learning theories, it is now possible to use a single parameter set for the segmentation of all annotations presented in the image. Additionally, there is potential for further research in the area of language-guided segmentation models, which would provide a more precise and flexible interface for medical professionals. We find building a model that incorporates these innovations intriguing.

We also noted that our model was trained in a self-supervised manner, meaning it has potentially gained a strong understanding of the structural features of ultrasonic images. This understanding is beneficial for downstream models such as the object detection model. Different ways of fine-tuning, like Low-Rank Adaptation (LoRA), adapter layers, etc. should be explored to find the optimal method to effectively transfer this understanding. We plan to address these issues in future studies.

Author Contributions

All authors contributed to the study's conception and design. Y.T. developed initial experiment setting. Z.X. refined the experiment details. Material preparation and data collection were performed by N.J., J.C. The first draft of the manuscript and the codebase was written by Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the Natural Science Foundation of Liaoning Province (Grant numbers 2022-MS-114). It was also supported by the Key R&D Plan Projects of Liaoning Province (Grant numbers 2020JH2/10300122).

Institutional Review Board Statement

No ethical approval is needed for this study, as it does not involve any human or animal subjects.

Informed Consent Statement

No consent is needed for this study, as it does not involve any human subjects.

Data Availability Statement

The data that support the findings of this study are available, upon reasonable request, from the corresponding authors. The data are not publicly available due to their containing information that could compromise the privacy of the patients to whom these ultrasound imaging results pertain. We released our code at https://github.com/ZhangYH-Z1RZcigZrw78AD-TR59O/UltrasonicImage-N2N-Approach.

Conflicts of Interest

The authors have no relevant financial or non-financial interests to disclose.

References

1. Kulshrestha, A.; Singh, J. Inter-hospital and intra-hospital patient transfer: Recent concepts. *Indian J. Anaesth.* **2016**, *60*, 451–457.
2. Jackson, P.; Chenal, C. Ultrasonic Imaging System with Body Marker Annotations. Google Patents. US Patent 9,713,458, 25 July 2017.
3. Li, H.; Xiong, P.; An, J.; Wang, L. Pyramid attention network for semantic segmentation. *arXiv* **2018**, arXiv:1805.10180.
4. Huang, Q.; Xia, C.; Wu, C.; Li, S.; Wang, Y.; Song, Y.; Kuo, C.-C.J. Semantic segmentation with reverse attention. *arXiv* **2017**, arXiv:1707.06426.

5. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144.
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. *Ssd: Single Shot Multibox Detector*; Springer: Amsterdam, The Netherlands, 2016; pp. 21–37.
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
8. Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; Aila, T. Noise2noise: Learning image restoration without clean data. *arXiv* **2018**, arXiv:1803.04189.
9. Kashyap, M.M.; Tambwekar, A.; Manohara, K.; Natarajan, S. Speech denoising without clean training data: a noise2noise approach. *arXiv* **2021**, arXiv:2104.03838.
10. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
11. Minaee, S.; Boykov, Y.Y.; Porikli, F.; Plaza, A.J.; Kehtarnavaz, N.; Terzopoulos, D. Image segmentation using deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022***, 44,* 3523–3542.
12. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
13. Chaurasia, A.; Culurciello, E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; pp.1–4.
14. Li, R.; Zheng, S.; Zhang, C.; Duan, C.; Su, J.; Wang, L.; Atkinson, P.M. Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–13.
15. Iakubovskii, P. Segmentation Models Pytorch. Available online: https://github.com/qubvel/segmentation_models.pytorch (accessed on 17 April 2024).
16. Chlap, P.; Min, H.; Vandenberg, N.; Dowling, J.; Holloway, L.; Haworth, A. A review of medical image data augmentation techniques for deep learning applications. *J. Med. Imaging Radiat Oncol.* **2021**, *65*, 545–563.
17. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation, In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Springer: Berlin, Germany; pp. 234–241.
18. Zhou, Z.; Rahman Siddiquee, M.M.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In *Proceedings of the Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Granada, Spain, 20 September 2018*; Springer: Berlin, Germany, 2018; pp. 3–11.
19. Ibtehaz, N.; Rahman, M.S. Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neur. Netw.* **2020**, *121*, 74–87.
20. Ponomarenko, N.; Silvestri, F.; Egiazarian, K.; Carli, M.; Astola, J.; Lukin, V. On between-coefficient contrast masking of dct basis functions. In Proceedings of the Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA, 13–15 January 2007; Volume 4.
21. Tieleman, T.; Hinton, G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neur. Netw. Mach. Learn.* **2012**, *4*, 26–31.
22. Foundation, T.P. RMSprop. Available online: https://pytorch.org/docs/stable/generated/torch.optim.RMSprop.html#torch.optim.RMSprop (accessed on 12 October 2022).