MASTER THESIS

Improvement of Robustness to Noise for Medical Image Segmentation by using Self-Supervised Learning Approach

Supervisor	Associate Professor	Takio Kurita
Sub Supervisor	Professor	Junichi Miyao
Sub Supervisor	Professor	Hiroaki Mukaidani

Hiroshima University
Mater of Informatics and Data Science

M213834 Yuta Konishi

A thesis submitted for the degree of the master at February 2023

Abstract

Making the trained model robust to the distortions, such as pixel noises in medical image segmentation, is crucial. Recently, self-supervised learning (SSL) methods such as SimCLR, VICReg, and Barlow Twins are closely related to spectral methods such as Laplacian Eigenmaps, Multidimensional Scaling, etc. This means that SSL can construct features invariant to the perturbations introduced by data augmentations. Since invariant feature extraction is also fundamental in medical image segmentation, we proposed introducing SSL loss as a regularizer in U-Net for medical image segmentation in this paper. Pixel noise is applied to the training samples, and invariant features to such distortions are extracted in the hidden layer of U-Net. The effectiveness of the proposed approach is experimentally confirmed using the subset of Sunnybrook Cardiac Data (SCD) and Abdominal Organs segmentation dataset by CHAOS.

Contents

1 Introduction **1**

2 Related Works **3**
2.1 Neural Network . . . 3
2.1.1 Active Function . . . 4
2.1.2 Convolutional Neural Network (CNN) . . . 4
2.2 Image Segmentation . . . 5
2.2.1 U-Net . . . 6
2.2.2 Medical Image Segmentation . . . 6
2.3 Self-Supervised Learning (SSL) . . . 7
2.3.1 SimCLR . . . 7
2.3.2 Other SSL Methods . . . 8

3 Proposed Method **11**
3.1 Problem Definition . . . 11
3.2 Network Architecture . . . 11
3.3 Training Flow . . . 12

4 Experimental Details **15**
4.1 Datasets . . . 15
4.1.1 Subset of SCD . . . 15
4.1.2 Abdominal Organs segmentation dataset by CHAOS challenge 15
4.2 Experimental Parameters . . . 16
4.2.1 Distortions . . . 16
4.2.2 Learning Parameters . . . 16
4.2.3 Evaluation . . . 16

5 Results **19**
5.1 Baseline experimental results . . . 19
5.2 Experimental results using subset of SCD dataset . . . 19
5.3 Experimental results using Abdominal Organs segmentation dataset . 20
5.4 Additional Experiments . . . 23
5.4.1 Number of linear transformations of the SSL branch . . . 23
5.4.2 2 SSL branches . . . 25
5.4.3 Noise intensity . . . 26

5.4.4 SSL loss function: CosineSimilarity 27

6 Conclusions **29**

Chapter 1

Introduction

Medical image segmentation is used to identify the pixels of organs or lesions from medical images such as CT or MRI images and is regarded as one of the most critical tasks in medical image analysis [1]. Deep learning is now recognized as one of the best approaches for medical image segmentation [2]. Many network architectures, such as the fully convolutional neural network (FCN) [3] or U-Net [4], have been used to segment medical images.

U-Net [4] is one of the most well-known architectures for medical image segmentation. The encoder-decoder architecture is utilized, and skip connections between different stages of the network are introduced, as shown in Fig.2.3. Many researchers applied the U-Net base model for medical image segmentation[5, 6].

Invariant feature extraction is one of the central topics in machine learning and pattern recognition, and it is also essential in deep learning. The standard approach to making robust to unnecessary variations is to train a deep learning model by using many training samples that include all possible variations. There are some researches in which invariant features are extracted by using deep learning. For example, pose-invariant features are extracted using Convolutional Neural Networks (CNN) for pose-invariant face recognition [7]. Metric learning has also often been used for invariant feature extraction [8, 9]. Ueda et al. proposed an invariant feature extraction method using Gradient Reversal Layer (GRL) [10].

Self-Supervised Learning (SSL) is one of the most promising methods to learn data representations that generalize across downstream tasks [11]. Labels in the training samples are not required, but the knowledge of what makes some samples semantically close to others is trained. Usually, semantic similarity is constructed by augmenting the training samples through data augmentations.

One of the basic SSL methods is SimCLR (a simple framework for contrastive learning of visual representations) [12]. SimCLR learns representations by maximizing agreement between differently augmented views of the same sample via a contrastive loss in the latent space. Recently Balestriero et al. [11] demonstrate that SSL methods such as SimCLR [12], VICReg [13], and Barlow Twins [14] are closely related with the spectral methods such as Laplacian Eigenmaps, Multidimensional Scaling, etc. This means that SSL can extract features (embedding) invariant to

the perturbations introduced by data augmentations.

The invariant feature extraction is also fundamental in supervised learning. Ramyaa et al. proposed to combine Barlow Twins loss with the standard cross entropy loss for the supervised learning with CNN [15].

In this paper, we propose to use SSL loss as a regularizer in U-Net-based medical image segmentation. Pixel noise is applied to the training samples as the distortions to the medical images, and the invariant features to such distortions are extracted by introducing SSL loss in the hidden layers of the U-Net. To show the effectiveness of the proposed approach, we have performed experiments using the subset of Sunnybrook Cardiac Data (SCD) [16] and Abdominal organs segmentation dataset by CHAOS challenge[17].

The contributions of this paper are summarized as follows:

(1) SSL loss in the hidden layers of the U-Net is introduced to make the trained model for medical image segmentation robust to the pixel noises.

(2) The effectiveness of the proposed approach is experimentally confirmed using the subset of Sunnybrook Cardiac Data (SCD) [16] and Abdominal Organs segmentation dataset by CHAOS challenge[17].

Chapter 2

Related Works

2.1 Neural Network

A neural network [18] is a mathematical model of human neurons and their connections. These artificial neurons are called perceptrons [18], and three or more layers of them are called multi-layer perceptrons (MLP) [18] or neural networks. A neural network consists of all fully connected layers, and one perceptron can be represented by the following

$$y = f({}^t\boldsymbol{w}\boldsymbol{x} + b) \tag{2.1}$$

where $\boldsymbol{x}$, y, $\boldsymbol{w}$, b, and $f(\cdot)$ are input vector, output, weight vector, bias, and active function (explained in the 1.1.1 section) respectively. Many of these perceptrons are connected and overlap to form a neural network as shown in Figure 2.1, which enables complex calculations. The function f is an activation function.

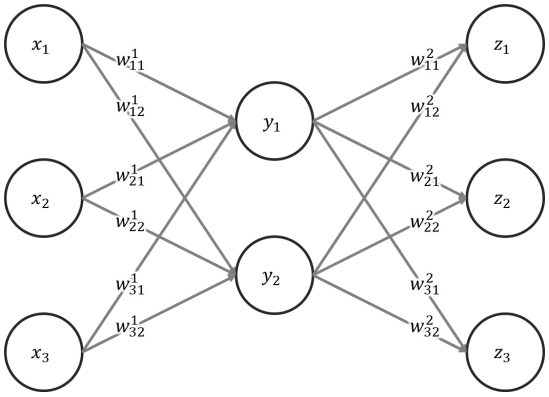

Figure 2.1: Overview of MLP.

2.1.1 Active Function

The activation function is a function that transforms the output values of a perceptron. This function can be inserted between a neural network to increase the flexibility of the representations of the model. In this study, we used the ReLU [19] and Softmax [20] functions.

ReLU

(Rectified Linear Unit) is one of the most commonly used activation functions of neural networks (figure 2.2). It outputs 0 if it is less than or equal to the input value, and the value as it is if it is greater than 0. This function is defined as

$$f(x) = \begin{cases} x & (x > 0) \\ 0 & (x \leq 0). \end{cases} \tag{2.2}$$

Softmax

is a function that normalizes the sum of multiple output values to be 1.0. This function is defined as

$$y_i = \frac{\exp(x_i)}{\sum_{k=1}^{n} \exp(x_k)} \qquad (i = 1, 2, ..., n) \tag{2.3}$$

where n is the number of classes.

2.1.2 Convolutional Neural Network (CNN)

A convolutional neural network (CNN) [18] is a neural network that includes a computation called convolution. It is mainly used for images and videos.

Convolution

is the most important structure in CNN. The convolution calculation generates new feature maps by filtering the input feature maps. The filter is also called a "kernel". The convolution operation is defined as

$$Output(x, y) = \sum_i \sum_j Filter(i, j) Input(x + i, y + j). \tag{2.4}$$

CNN performs this convolutional computation instead of a linear computation.

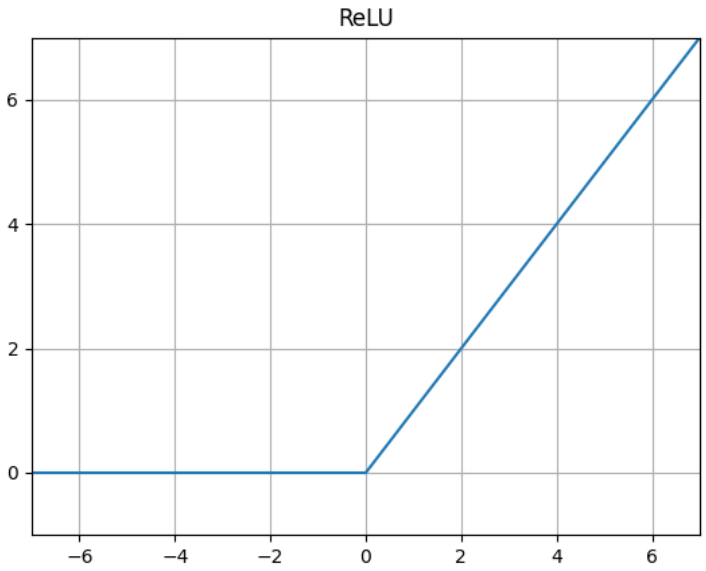

Figure 2.2: Graphs of ReLU.

Pooling

is a technique used to enhance and compress feature maps. As with convolution, the calculation is performed using a kernel. One of the most commonly used is MaxPooling, which retrieves the largest value in the kernel.

2.2 Image Segmentation

Image segmentation is one of the image recognition techniques and is the task of partitioning objects in an image. It has been applied in a wide range of fields such as medical image analysis, scene understanding, robot perception, video surveillance, augmented reality, image compression, automated driving, and so on [21].

There are three main segmentation methods: semantic segmentation, instance segmentation, and panoptic segmentation.

Semantic Segmentation is a method of attaching a class label to each pixel in an image. It divides the image into regions for each type of object in the image and performs class classification on a pixel-by-pixel basis.

Experiments have been conducted using this method in this study.

Instance Segmentation differs from semantic segmentation in that it is a method for segmenting objects in an image into individual regions. Because it does not segment by class, if two or more objects in an image are identical, they are recognized as different objects.

Panoptic Segmentation is a technique that combines semantic and instance segmentation described above. All objects in the image are assigned class labels, and

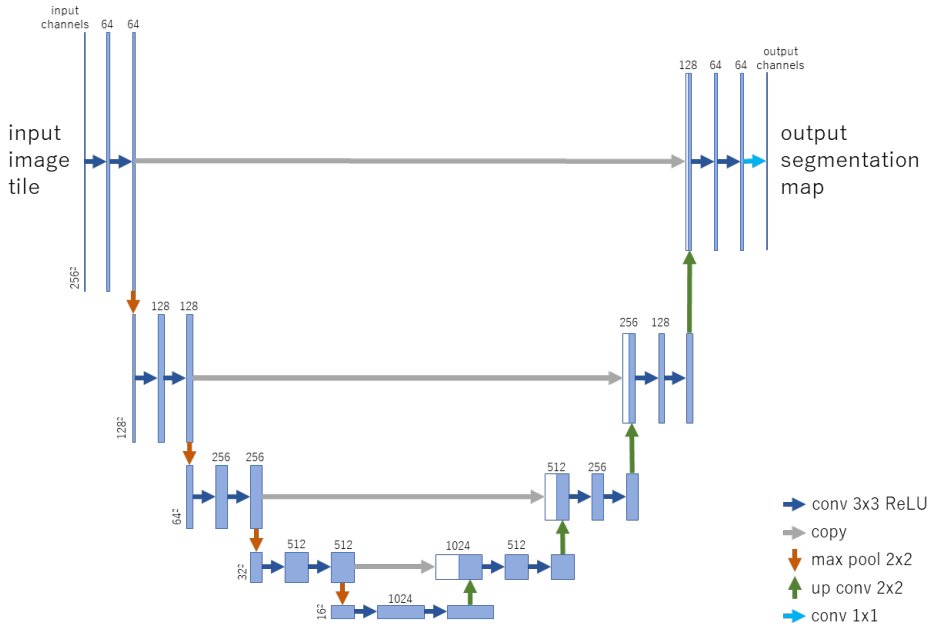

Figure 2.3: Overview of U-Net.

if there are multiple identical objects, they are recognized individually. Currently, it is not sufficiently accurate and is still a developing research field.

2.2.1 U-Net

U-Net is a network proposed by Ronneberger et al. [4] for biomedical image segmentation, one of the fully convolutional networks (FCN) [3] and currently the most popular network model used in semantic segmentation (figure 2.3). An FCN is a network composed entirely of convolutional layers, unlike CNNs, which are generally composed of a convolutional layer and a fully-connected layer [22]. While general CNN's output feature vectors can be classified, FCNs can output feature maps or images.

In addition to being an FCN, U-Net has the features of deconvolution and skip-connection. deconvolution is the opposite of convolution, which is an operation to stretch the input features (upsampling). Skip-connection is the process of adding the features obtained by convolution to those to be deconvolved. This allows the features related to the position of the object lost by pooling, etc. to be supplemented.

U-Net has an encoder-decoder architecture because it compresses the input image into a small feature map and restores it to the original image size.

2.2.2 Medical Image Segmentation

Medical image segmentation, which identifies pixels of organs and lesions from medical images such as CT and MRI, plays an important role in many medical image

analyses [23, 1]. Deep learning is now recognized as one of the best approaches for medical image segmentation [2].

While deep learning-based models produce good segmentation accuracy, it is essential to collect a large number of training samples to learn them [3]. In medical image analysis, the collection of large numbers of image samples is often a very difficult and expensive task. The most common approach to increase the amount of training samples is data augmentation, which applies variation to images in the training sample [24].

In medical images, it is common for anatomical structures of interest to occupy only a small portion of the image. That is, most pixels in the image belong to the background region, and the organs or lesions that should be seen for medical diagnosis are very small. When a network is trained on such data, the learned network is often biased toward the background. A common solution to this problem is sample reweighting, which applies a higher weight to foreground patches. Dice loss is often used for automatic reweighting [25, 4, 26, 27].

Another approach is to introduce prior knowledge into the loss function as a regularization. For example, the Euler characteristic (EC) from topology computes the number of isolated objects on segmented vascular regions in the fundus image and uses it as a regularization factor for training [28]. It is also useful to use the information on pixel neighborhood relationships; Hakim et al. [29] proposed to introduce a regularization term defined based on the difference of neighboring pixels. The regularization term can be expressed as a graph Laplacian computed from the output of the network and the ground-truth image.

2.3 Self-Supervised Learning (SSL)

Recently, it has been shown that self-supervised learning (SSL) can extract as many features as supervised learning with a large number of training samples [11]. SSL can construct a representation of unlabeled data, which has led to significant advances in a variety of applications, including natural language processing, speech processing, and computer vision [30]. SSL can build representations of unlabeled data and has led to significant advances in various applications such as natural language processing, speech processing, and computer vision.

In SSL for computer vision, distortions and variations are added to the original image. The features of the distorted images are then trained to be close to each other. This is accomplished by maximizing the similarity of representations obtained with different distortions using deformation of the Siamese network [31]. In this way, SSL is able to learn representations (embeddings) that are invariant to distortions applied to the input image.

2.3.1 SimCLR

SimCLR (a simple framework for contrastive learning of visual representations) is one of the basic methods for contrastive self-supervised learning [12]. SimCLR learns

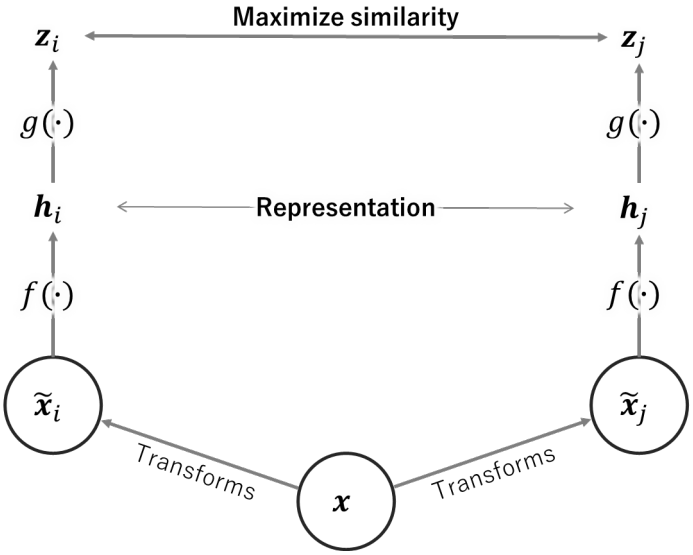

Figure 2.4: Overview of SimCLR.

representations by maximizing agreement between differently augmented views of the same sample via a contrastive loss in the latent space (figure 2.4).

Let $\{\boldsymbol{x}_k|k = 1, \ldots, N\}$ be the training samples in a mini-batch. At first, for each training sample in the mini-batch, a stochastic data augmentation is applied to randomly generate two views of the same sample, denoted $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$, which are considered as a positive pair. Then we obtain $2N$ pairs of the augmented samples derived from the samples in the mini-batch. These augmented samples include a positive pair $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$ which are generated from the same training sample $\boldsymbol{x}_i$. The pairs of the augmented samples are fed into the neural network encoder to get the hidden representation $\boldsymbol{h}_i = f(\tilde{\boldsymbol{x}}_i$. The contrastive loss is applied after the hidden representation is mapped by a small neural network projection head as $\boldsymbol{z}_i = g(\boldsymbol{h}_i)$.

The loss function for a positive pair of examples (i, j) is defined as

$$l_{i,j} = -log\frac{\exp(sim(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}_{[k\neq i]} \exp(sim(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)} \tag{2.5}$$

where $sim(\boldsymbol{u}, \boldsymbol{v}) = \dfrac{\boldsymbol{u}^T\boldsymbol{v}}{||\boldsymbol{u}||||\boldsymbol{v}||}$ is the cosine similarity between two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ and $\mathbf{1}_{[k\neq i]}$ is an indicator function evaluating to 1 if $k \neq i$. τ denotes a temperature parameter that controls the scale. Then the final loss is computed across all positive pairs in the mini-batch.

2.3.2 Other SSL Methods

For SSL, it is important to prevent a collapse in which the encoders produce constant or non-informative representations. Bardes et al. proposed VICReg (Variance-Invariance-Covariance Regularization) which explicitly avoids the collapse problem

with two regularization terms [13]. One term maintains the variance of each embedding dimension above a threshold and the other decorrelates each pair of variables.

Another method is Barlow Twins which is developed by applying H. Barlow ' s redundancy-reduction principle [14]. The objective function of Barlow Twins measures the cross-correlation matrix between the embeddings of two identical networks fed with distorted versions of a batch of samples and tries to make this matrix close to the identity matrix. This makes the embedding vectors of distorted versions to be similar while minimizing the redundancy between the components of these vectors. It is reported that Barlow Twins is competitive with state-of-the-art methods for SSL.

Balestriero et al. [11] demonstrate that SSL methods such as SimCLR, VICReg, and Barlow Tiwns correspond to the spectral methods such as Laplacian Eigenmaps, Multidimensional Scaling, etc. This shows that invariant feature extraction is fundamental in SSL.

Since it is obvious that the invariant feature extraction is also important in supervised learning, Barlow Twins loss is combined with the standard cross-entropy loss as a regularizer in the supervised learning with CNN [15].

In this paper, we propose to introduce SSL loss as a regularizer in U-Net for medical image segmentation.

Chapter 3

Proposed Method

3.1 Problem Definition

In medical image segmentation, the accuracy of segmentation is significantly reduced if the medical image contains pixel noise. This is because the boundary between the background and the organ or lesion to be marked becomes ambiguous.

Therefore, we designed a mechanism to promote segmentation learning that is robust to pixel noise in medical images. As described in the previous section, SSL is capable of learning representations that are invariant to distortions given the image. We exploit this property by learning the segmentation task and SSL in parallel, and by introducing the SSL loss function as a regularization term in the segmentation loss function during segmentation learning to facilitate segmentation learning to be robust to distorted images.

3.2 Network Architecture

We propose a mechanism that connects a network consisting of 2 linear layers to the intermediate layer of the segmentation model and uses the output feature vector for learning SSL (figure 3.1).

U-Net is used to train the main task, segmentation. It takes a grayscale medical image as input, goes through encoding and decoding, and outputs a feature map (pixel-by-pixel probability distribution) of the same size as the input.

To train the SSL for normalization, the intermediate features of the U-Net are input to a network consisting of two linear layers (SSL branch) and its output feature vector is used. The SSL branch follows SimCLR and performs two linear transformations [12].

$$z_i = g(\boldsymbol{h}_i) = W^{(2)}\sigma(W^{(1)}\boldsymbol{h}_i) \tag{3.1}$$

where $h, W, \sigma, g(\cdot)$, and z are the intermediate features of the U-Net, the weights of the linear layer, the ReLU function, the two-layer linear network, and the final output of the SSL branch, respectively (figure 2.4). The SSL branch can be connected to each block (encode1-5, decode1-3) as shown in the figure 3.1, from which

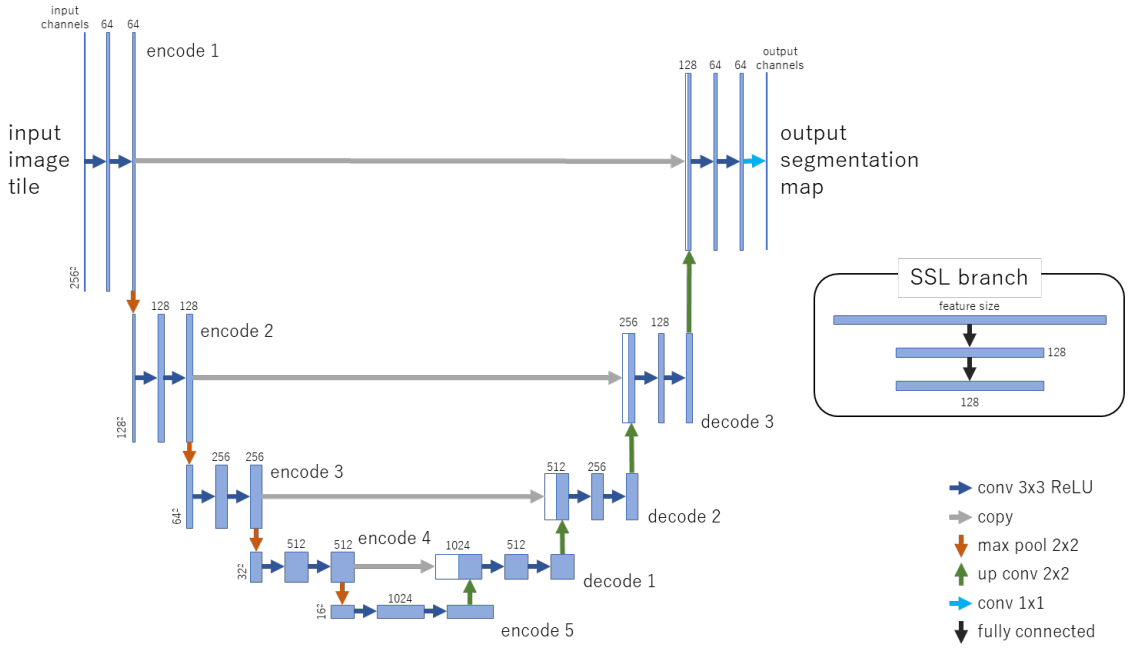

Figure 3.1: Overview of proposed network.

the intermediate features are transformed into a 128-dimensional feature vector by two linear layers. We expect the U-Net to embed an invariant representations of the images by learning to approach the final output vectors of the positive paired samples by SSL branch.

3.3 Training Flow

Follow the learning method used in SimCLR's paired data learning. The original image is subjected to Gaussian noise, and the distorted images are used as positive paired samples with the original image.

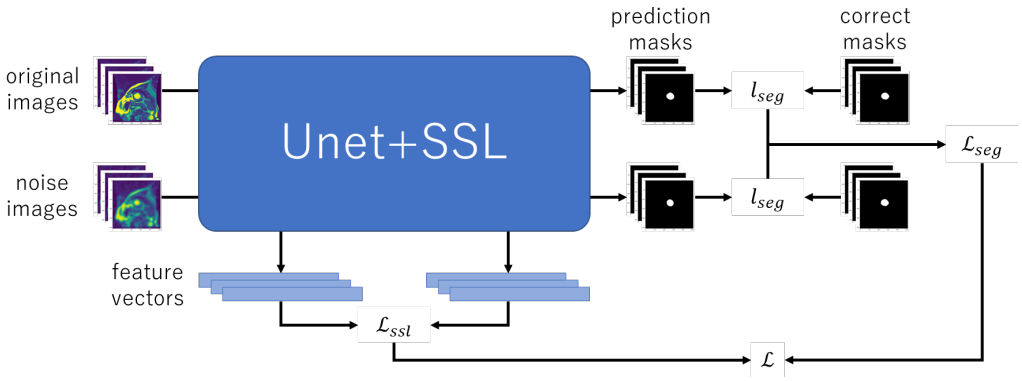

Figure 3.2: Training flow of the proposed method.

The original and the distorted images are fed to the main stream (U-Net), and the outputs of the U-Net are used to compute the segmentation loss function for each image. The SSL branch of the sub-stream outputs the feature vectors of the original and the distorted images, and these vectors are used to compute the SSL loss function. This allows the model to adapt to the effect of bridging the difference between the positive paired samples of the original and distorted images when learning the segmentation task. As a result, the model can incorporate representations that are invariant to variation (noise) and robust to such distortions. The overview of the training flow of the proposed method is shown in the figure 3.2.

In the proposed learning flow, Segmentation learning and SSL are performed simultaneously. This means that the loss functions must be computed and fused. In this study, the loss function is defined as the weighted sum of the loss functions of segmentation and SSL as

$$\mathcal{L} = \lambda \mathcal{L}_{seg} + (1 - \lambda)\mathcal{L}_{ssl} \tag{3.2}$$

where $\mathcal{L}_{seg}$ and $\mathcal{L}_{ssl}$ are the loss functions of segmentation and SSL and λ is the coefficients that determine the ratio of the two loss functions. Since the segmentation loss function is computed for each of the original and the distorted images, the segmentation loss $\mathcal{L}_{seg}$ is defined by their respective averages as

$$\mathcal{L}_{seg} = \frac{l_{seg}(Y_{original}) + l_{seg}(Y_{noise})}{2} \tag{3.3}$$

where l_{seg} is the loss function of segmentation and $Y_{original}$ and Y_{noise} are the outputs of U-Net for the original and the distorted images. In this study, Cross-entropy Loss is used for segmentation loss and InfoNCE Loss (eq (2.5)) is used for SSL loss.

Chapter 4

Experimental Details

4.1 Datasets

To evaluate the effectiveness of the proposed approach, we have performed experiments using two datasets. They are the subset of Sunnybrook Cardiac Data (SCD) [16] and Abdominal Organs segmentation dataset by CHAOS challenge[17]. Images of the datasets are resized to 256×256 pixels.

4.1.1 Subset of SCD

The SCD also called the 2009 Cardiac MR Left Ventricular Segmentation Challenge data, consists of 45 cine MRI images of various patients and conditions. The SCD subset used in this study consists of gray-scale cardiac MRI images (short-axis images) and expert-masked data of the left ventricular region (figure 4.1). The masked data is a binary image with 1 inside the region of the left ventricle and 0 in other regions. The training data set consists of 234 image pairs, and the validation data set consists of 26 image pairs. They do not overlap each other.

4.1.2 Abdominal Organs segmentation dataset by CHAOS challenge

The CHAOS Challenge is aimed at segmenting organs (liver, kidneys, spleen) from abdominal CT and MRI data. CT and MRI are provided in DICOM image data, each with masked images of abdominal organs. The CT dataset is data acquired for the pre-evaluation of living liver transplant donors and is intended for the segmentation of the liver. The MRI data set consists of data from two different sequences (T1-DUAL and T2-SPIR) and is intended for the segmentation of the four abdominal organs (liver, right and left kidneys, and spleen). The MRI T2-SPIR data set was used in this experiment (figure 4.2). As mentioned earlier, this data set is DICOM image data, so it was converted to JPEG image data for easier handling. Of the total MRI images, 531 were used as training data and 92 as validation data. The

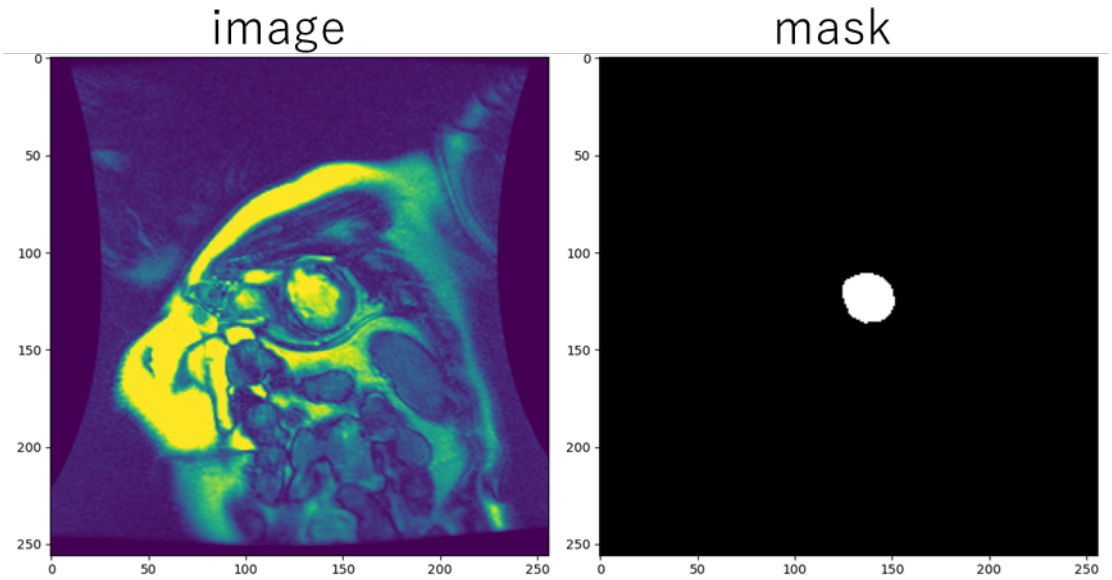

Figure 4.1: Image and mask in subset of SCD dataset.

classes to be classified are the four abdominal organs (liver, right and left kidneys, and spleen) as described above.

4.2 Experimental Parameters

4.2.1 Distortions

The distortion used in this study is Gaussian noise. Gaussian noise is statistical noise that has the same probability density function as the Gaussian distribution. The noise image was generated by adding 0.3 times the noise to the original image.

4.2.2 Learning Parameters

The batch size was set to 9, and Adam was used as the optimizer.

For the subset of SCD, the number of epochs was set to 100, and the learning rate was set to 0.0001, which was multiplied by 0.5 every 25 epochs. The weight decay was set to 0.001.

For the Abdominal Organs segmentation dataset, the number of epochs was set to 250, and the learning rate was set to 0.0001, which was multiplied by 0.5 every 40 epochs. The weight decay was set to 0.01.

4.2.3 Evaluation

Multi-class IoU and pixel-wise accuracy, which are common metrics for segmentation tasks, were used for evaluation. After training the model, prediction using the

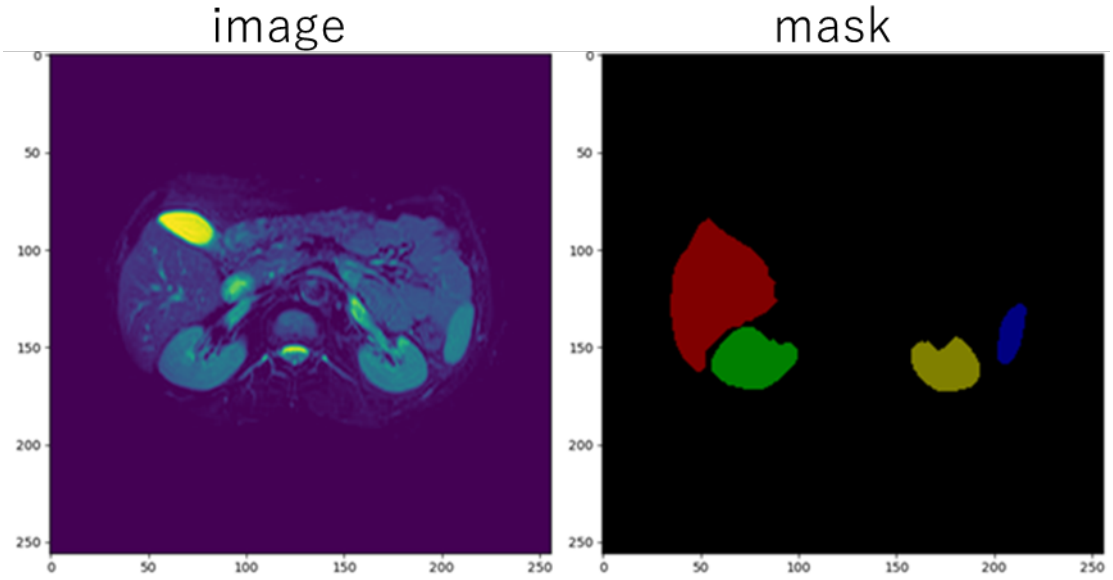

Figure 4.2: Image and mask in Abdominal Organs segmentation dataset.

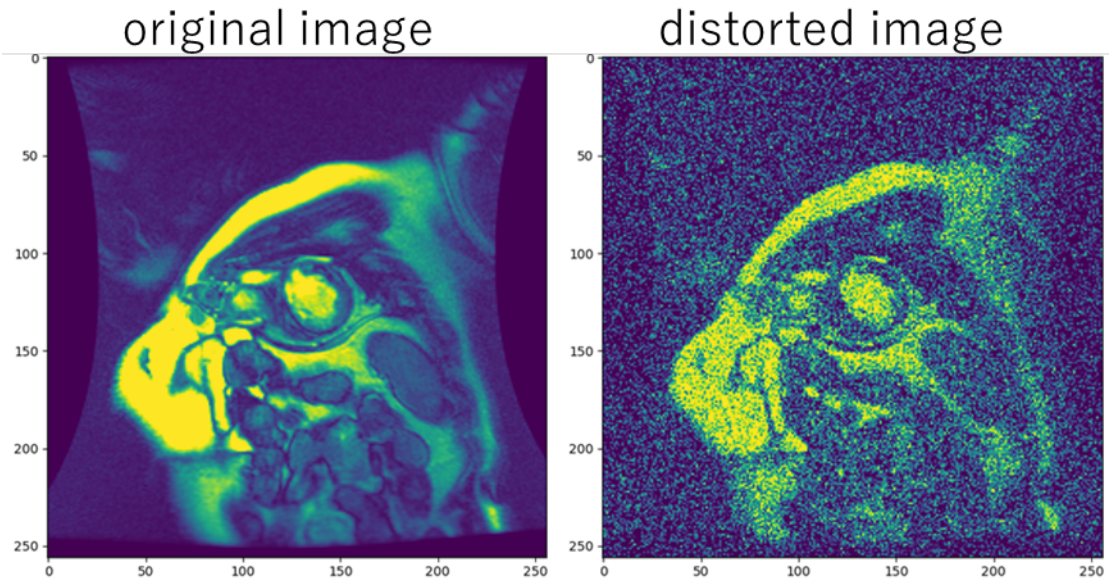

Figure 4.3: Original and distorted images in subset of SCD dataset.

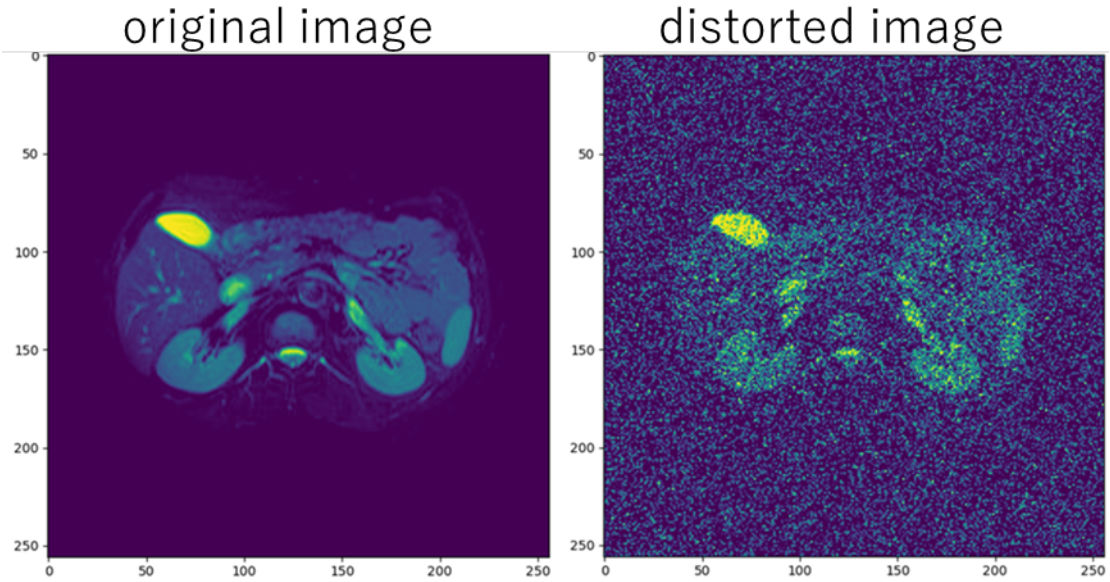

Figure 4.4: Original and distorted images in Abdominal Organs segmentation dataset.

trained model is performed on the original images and the distorted images, and each is evaluated.

Chapter 5

Results

5.1 Baseline experimental results

First, check the segmentation accuracy of the distorted images by baseline. The U-Net was trained using only the original images as baseline1. Also, let baseline2 be the one in which the U-Net is trained on a dataset in which 50% of the samples are replaced by distorted images. We will use these baselines as a baseline to check the performance of the proposed method.

Table 5.1: Baseline accuracy comparison for subset of SCD.

	IoU		pixel-wise accuracy	
	original images	distorted images	original images	distorted images
baseline1	94.625	49.115	99.810	98.230
baseline2	94.284	93.531	99.799	99.772

Table 5.2: Baseline accuracy comparison for Abdominal Organs segmentation dataset.

	multi-class IoU		pixel-wise accuracy	
	original images	distorted images	original images	distorted images
baseline1	87.329	19.102	99.355	95.511
baseline2	82.570	80.478	99.019	98.855

5.2 Experimental results using subset of SCD dataset

We present the results of the proposed method using a subset of the SCD dataset.

The proposed method has two parameters: the location of the SSL branches connected to the U-Net and a λ that determines the fraction of segmentation and SSL loss function(3.2). Since these two parameters are expected to influence each other, we tried one combination of parameters in this experiment. Specifically, 72 experiments were conducted, including 8 different connection positions of the SSL

branch (encode1-5, decode1-3) and 9 different values of λ, varying by 0.1 from 0.1 to 0.9 in 0.1 steps.

The results for each evaluation indicator for the test data are as follows.

Table 5.3: multi-class IoU using original images from subset of SCD dataset

λ	encode1	encode2	encode3	encode4	encode5	decode1	decode2	decode3
0.1	**94.665**	93.797	**95.379**	91.104	91.465	91.868	91.219	88.785
0.2	**94.661**	**95.072**	**95.110**	**95.026**	92.799	92.844	92.616	90.295
0.3	**95.441**	**95.492**	**95.313**	94.302	92.291	94.043	92.119	91.682
0.4	94.425	**95.471**	**95.341**	**94.743**	94.125	94.233	93.383	89.340
0.5	**95.243**	**95.546**	**95.539**	**94.924**	93.923	94.045	93.187	94.195
0.6	**94.996**	**95.457**	**95.530**	**95.045**	93.688	94.158	94.524	93.992
0.7	**95.462**	**95.463**	**95.451**	94.359	93.903	94.379	**95.016**	94.284
0.8	**95.293**	***95.821***	**95.537**	94.441	**95.158**	93.952	**94.959**	**95.192**
0.9	**95.685**	**95.544**	**95.462**	**94.797**	94.566	94.546	94.502	**95.469**

Table 5.4: multi-class IoU using distorted images from subset of SCD dataset

λ	encode1	encode2	encode3	encode4	encode5	decode1	decode2	decode3
0.1	**94.207**	92.387	93.422	90.360	85.980	86.537	85.080	86.101
0.2	92.449	92.285	93.387	92.779	86.892	88.300	89.340	85.258
0.3	**93.934**	92.513	**94.490**	93.067	88.438	91.574	89.812	87.630
0.4	**93.872**	**93.847**	**94.968**	92.366	89.531	91.108	92.264	84.128
0.5	93.519	**94.898**	**94.696**	93.010	91.339	90.989	91.950	91.666
0.6	93.465	**94.693**	**94.627**	93.247	91.178	90.927	92.087	92.410
0.7	**94.074**	**95.158**	**94.946**	93.279	93.421	92.334	92.056	92.444
0.8	**94.635**	***95.336***	**94.974**	93.031	**93.754**	93.040	**93.672**	93.314
0.9	**94.340**	**95.110**	**94.651**	**94.067**	**94.024**	**93.943**	**93.654**	93.277

When compared to the larger value of each baseline evaluation value, the better evaluation value of the proposed method is in bold, and furthermore, the most accurate one is in italics.

Table (5.3,5.4,5.5,5.6) of the experimental results shows that the best accuracy is obtained when the SSL branch is connected to encode2 and the λ is set to 0.8 for both the original and distorted images. This indicates that the contour information of objects in the image could be extracted from the upper layers of U-Net and robust to the object contours.

The proposed method is also more accurate for larger values of the parameter λ, indicating that it is robust to small changes in λ.

5.3 Experimental results using Abdominal Organs segmentation dataset

Next, we show the experimental results from the Abdominal Organs segmentation dataset.

Table 5.5: pixel-wise accuracy using original images from subset of SCD dataset

λ	encode1	encode2	encode3	encode4	encode5	decode1	decode2	decode3
0.1	99.810	99.780	**99.832**	99.650	99.694	99.699	99.678	99.592
0.2	**99.812**	**99.825**	**99.822**	**99.821**	99.746	99.732	99.734	99.624
0.3	**99.837**	**99.839**	**99.829**	99.789	99.726	99.786	99.716	99.682
0.4	99.802	**99.838**	**99.829**	99.808	99.788	99.792	99.754	99.564
0.5	**99.830**	**99.840**	**99.838**	**99.816**	99.782	99.784	99.751	99.783
0.6	**99.822**	**99.836**	**99.837**	**99.823**	99.770	99.787	99.803	99.777
0.7	**99.838**	**99.836**	**99.835**	99.797	99.774	99.796	**99.820**	99.787
0.8	**99.831**	***99.850***	**99.838**	99.802	**99.824**	99.777	**99.820**	**99.826**
0.9	**99.845**	**99.839**	**99.835**	**99.814**	99.803	99.802	99.803	**99.836**

Table 5.6: pixel-wise accuracy using distorted images from subset of SCD dataset

λ	encode1	encode2	encode3	encode4	encode5	decode1	decode2	decode3
0.1	**99.793**	99.731	99.765	99.657	99.508	99.510	99.468	99.498
0.2	99.732	99.729	99.763	99.744	99.542	99.581	99.616	99.469
0.3	**99.784**	99.735	**99.802**	99.750	99.593	99.694	99.639	99.558
0.4	**99.782**	**99.781**	**99.819**	99.729	99.626	99.683	99.720	99.413
0.5	99.769	**99.817**	**99.808**	99.752	99.693	99.673	99.702	99.692
0.6	99.767	**99.810**	**99.806**	99.759	99.685	99.677	99.717	99.725
0.7	**99.787**	**99.825**	**99.817**	99.760	99.764	99.728	99.719	99.722
0.8	**99.808**	***99.832***	**99.818**	99.753	**99.775**	99.750	**99.776**	99.761
0.9	**99.798**	**99.824**	**99.806**	**99.788**	**99.785**	**99.781**	**99.775**	99.750

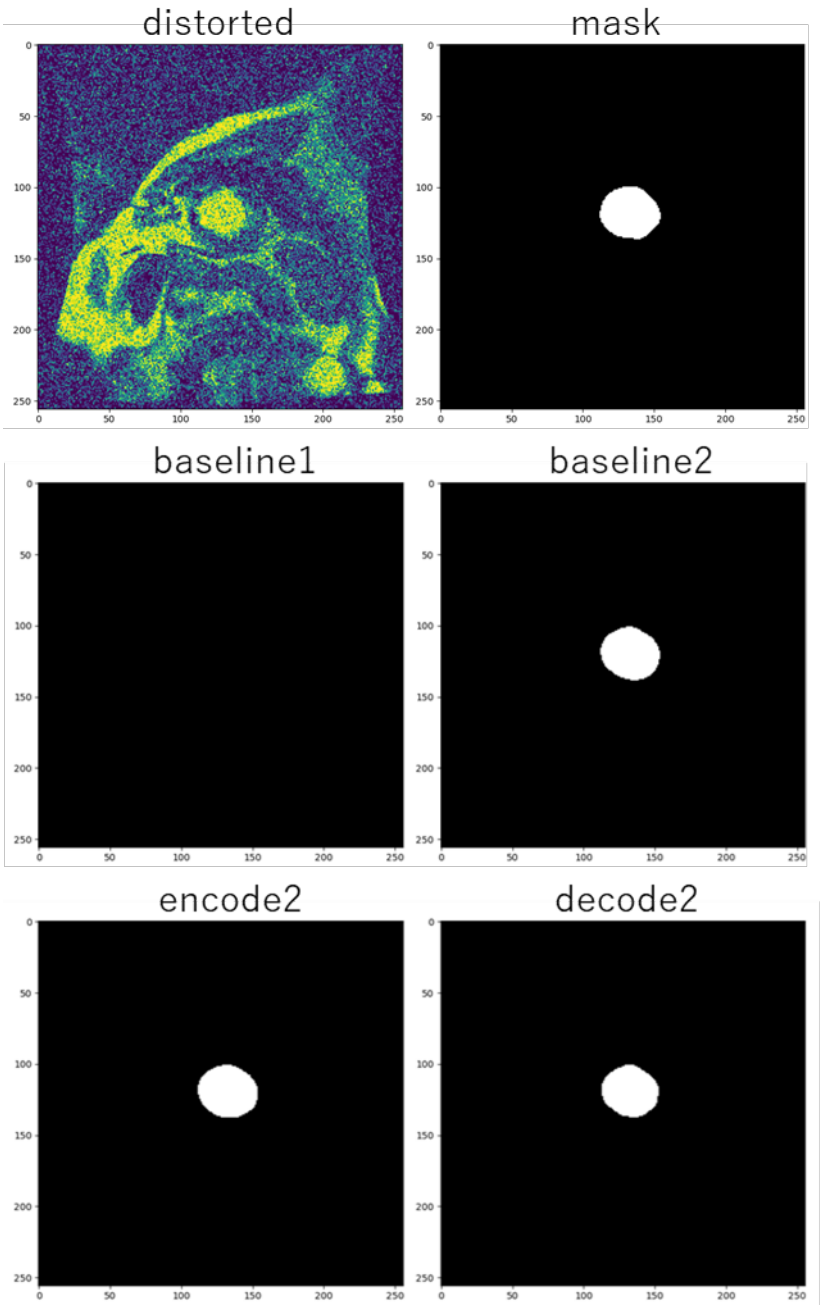

Figure 5.1: Result image of segmentation by each methods with subset of SCD dataset.

Based on the results of the SCD dataset experiment described in the previous section, we connected the SSL branch to two locations, encode2, which had the best performance, and decode2, which had consistently good performance on the decoder side, and performed the experiment with the λ set to 0.8.

Table 5.7: Comparison of accuracy with Abdominal Organs segmentation dataset.

	multi-class IoU		pixel-wise accuracy	
model	original images	distorted images	original images	distorted images
baseline1	**87.329**	19.102	**99.355**	95.511
baseline2	82.570	80.478	99.019	98.855
encode2	82.231	80.393	99.076	***98.938***
decode2	83.723	***82.137***	99.059	**98.932**

Table 5.7 shows that the proposed method outperforms baseline 1 for distorted images, while all of them are lower than baseline 1 for the original images. The best accuracy is obtained when the SSL branch is connected to decode2 when measured by multi-class IoU and to encode2 when measured by pixel-wise accuracy (however, pixel-wise accuracy for distorted images when connected to decode2 also exceeds baseline, and the values are close). accuracy also exceeds baseline and the values are close). This indicates that in the case of multi-class segmentation, it is easier to improve performance by extracting class information from the lower layers of the U-Net than by extracting contour information of objects in the image from the upper layers of the U-Net.

5.4 Additional Experiments

5.4.1 Number of linear transformations of the SSL branch

In this study, following SimCLR, the number of linear transformations of the SSL branch is performed twice. However, we will check how the proposed method changes when this number of transformations is reduced to one. We set the lambda to 0.8 and evaluate the performance on two datasets when the SSL branch is connected to encode2 and decode2, respectively.

Table 5.8: Comparison by number of linear transformations with subset of SCD dataset.

		multi-class IoU		pixel-wise accuracy	
model		original images	distorted images	original images	distorted images
baseline1		94.625	49.115	99.810	98.230
baseline2		94.284	93.531	99.799	99.772
fc×1	encode2	**95.118**	**94.627**	**99.824**	**99.808**
	decode2	**94.649**	93.491	99.806	99.763
fc×2	encode2	***95.821***	***95.336***	***99.850***	***99.832***
	decode2	**94.959**	**93.672**	**99.820**	**99.776**

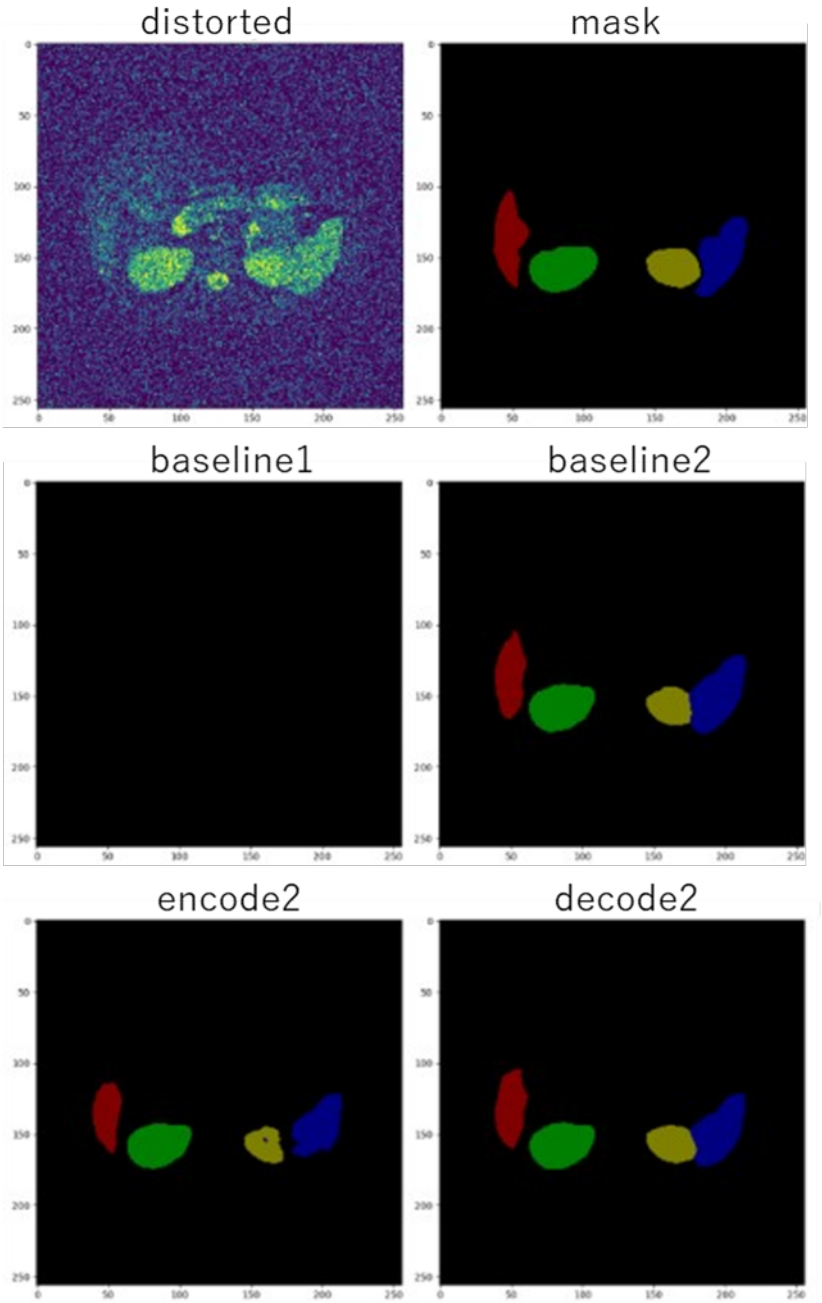

Figure 5.2: Result image of segmentation by each methods with Abdominal Organs segmentation dataset.

Table 5.9: Comparison by number of linear transformations with Abdominal Organs segmentation dataset.

model		multi-class IoU		pixel-wise accuracy	
		original images	distorted images	original images	distorted images
baseline1		**97.329**	19.102	**99.355**	95.511
baseline2		82.570	80.478	99.019	98.855
fc×1	encode2	81.087	79.186	99.028	**98.855**
	decode2	84.162	**81.310**	99.082	**98.893**
fc×2	encode2	82.231	80.393	99.076	***98.938***
	decode2	83.723	***82.137***	99.059	**98.932**

From Table 5.8, 5.9, it can be seen that in most cases, the performance is worse when the number of linear transformations is one than when it is two. Therefore, the number of linear transformations will be performed twice in subsequent experiments.

5.4.2 2 SSL branches

To see what happens to performance when two SSL branches are connected and trained at the same time. Connect SSL branches to encode2 and decode2 and proceed with training. However, the SSL loss function $\mathcal{L}_{ssl}$ is the average of the loss functions of encode2 and decode2.

Table 5.10: Comparison by SSL branches with subset of SCD dataset.

model	multi-class IoU		pixel-wise accuracy	
	original images	distorted images	original images	distorted images
baseline1	49.625	49.115	99.810	98.230
baseline2	94.284	93.531	99.799	99.772
encode2	***95.821***	***95.336***	***99.850***	***99.832***
decode2	**94.959**	**93.672**	**99.820**	**99.776**
encode2+decode2	**95.261**	**94.789**	**99.826**	**99.810**

Table 5.11: Comparison by SSL branches with Abdominal Organs segmentation dataset.

model	multi-class IoU		pixel-wise accuracy	
	original images	distorted images	original images	distorted images
baseline1	**97.329**	19..102	**99.355**	95.511
baseline2	82.570	80.478	99.019	98.855
encode2	82.231	80.393	99.076	***98.938***
decode2	83.723	***82.137***	99.059	**98.932**
encode2+decode2	76.819	75.337	98.799	98.675

Table 5.10, 5.11 shows that the performance when two SSL branches are connected tends to be lower than that when one SSL branch is connected, respectively. Although we conducted the experiment with the expectation that feature extraction

from two locations would synergistically improve performance, we could not confirm any improvement in performance in this experiment. However, we did not see any improvement in performance in this experiment. It may be possible to improve performance a little more by tuning parameters.

5.4.3 Noise intensity

We tested how the strength of the distorted images (Gaussian noise) applied to the image data during training changes the performance of the model after training. We trained models with noise strengths of 0.1, 0.3, 0.5, 0.7, and 0.9, and evaluated them on images with the same noise strength. The SSL branch was connected to encode2, lambda was set to 0.8, and a subset of the SCD dataset was used.

Table 5.12: Performance variation with image noise intensity in training the proposed method with subset of SCD dataset (multi-class IoU). The rows are the noise intensity applied to the image during training and the columns are the noise intensity applied to the image during evaluation.

	0.0	0.1	0.2	0.3	0.4	0.9
0.1	95.970	95.773	51.951	49.115	49.115	49.115
0.3	95.458	95.544	95.322	89.317	62.539	50.153
0.5	95.354	95.327	95.399	94.752	91.532	82.604
0.7	94.999	94.827	94.921	94.703	93.288	87.649
0.9	93.914	94.206	94.662	94.375	93.626	91.325

Table 5.13: Performance variation with image noise intensity in training the proposed method with subset of SCD dataset (pixel-wise accuracy). The rows are the noise intensity applied to the image during training and the columns are the noise intensity applied to the image during evaluation.

	0.0	0.1	0.3	0.5	0.7	0.9
0.1	99.854	99.848	98.329	98.230	98.230	98.230
0.3	99.836	99.838	99.830	99.622	98.694	98.267
0.5	99.833	99.831	99.833	99.809	99.697	99.389
0.7	99.821	99.814	99.816	99.807	99.757	99.561
0.9	99.783	99.792	99.805	99.792	99.764	99.684

The table 5.12, 5.13 is colored according to the magnitude of the values. In the table, the largest values are colored green, the smallest values are colored red, and values in between are represented by gradients.

Table 5.12, 5.13 shows that the larger the noise applied during training, the lower the accuracy with respect to the original image, but the generalization performance for noise of various strengths is improved.

5.4.4 SSL loss function: CosineSimilarity

In this study, InfoNCE Loss (eq (2.5)) is used for the SSL loss function. This function has the effect of bringing the features of positive pairs (the original image and its distorted images) in a mini-batch closer together and moving the other features away. Since regular SSL is positioned as a pre-training for the class classification task, the effect of moving away non-positive pairs is necessary. However, in the case of segmentation tasks, even different images may contain objects belonging to the same class, so the effect of moving away non-positive pairs is not considered necessary.

Therefore, we conducted an additional experiment in which the SSL loss function was changed to Cosine Similarity Loss. Cosine Similarity Loss is a loss function that learns to maximize cosine similarity. In other words, only the effect of approaching positive pairs of features can be obtained during learning. Two datasets were used to compare the results with those of baseline, the proposed method (InfoNCE Loss).

Table 5.14: Variation in performance with different loss functions using subset of the SCD dataset.

model		multi-class IoU		pixel-wise accuracy	
		original images	distorted images	original images	distorted images
baseline1		94.625	49.115	99.810	98.230
baseline2		94.284	93.531	99.799	99.772
InfoNCE	encode2	***95.821***	***95.336***	***99.850***	***99.832***
	decode2	**94.959**	**93.672**	**99.820**	**99.776**
CosineSim	encode2	**95.647**	**93.598**	**99.844**	99.764
	decode2	**95.504**	**94.568**	**99.838**	**99.806**

Table 5.15: Variation in performance with different loss functions using Abdominal Organs segmentation dataset.

model		multi-class IoU		pixel-wise accuracy	
		original images	distorted images	original images	distorted images
baseline1		**97.329**	19.102	**99.355**	95.511
baseline2		82.570	80.478	99.019	98.855
InfoNCE	encode2	82.231	80.393	99.076	***98.938***
	decode2	83.723	**82.137**	99.059	**98.932**
CosineSim	encode2	81.136	77.318	99.094	98.763
	decode2	83.011	***82.226***	99.083	**98.885**

From the table, the results did not exceed the InfoNCE Loss results in most cases, although they could exceed the baseline. This result indicates that in the segmentation task, it is necessary to act not only to bring the features closer together, but also to move them apart.

Chapter 6

Conclusions

We proposed a learning method for U-Net with SSL to make the trained model robust against image distortions such as pixel noise. The proposed method (U-Net with SSL) can construct the segmentation model by extracting features that are invariant to distortions in the paired data. The effectiveness of the proposed approach was experimentally confirmed by using the subset of Sunnybrook Cardiac Data (SCD) and Abdominal Organs segmentation dataset.

In this paper, we used only pixel noise as image distortion. We think the approach proposed in this paper can apply to the other types of image distortions. Experiments for such distortions will be our future works.

Acknowledgements

I want to thank Professor Takio Kurita. Professor Kurita gave me many ideas and advice for this research. Without his guidance, this research would not have been possible.

I am also grateful to my friends, seniors, and juniors in the laboratory. Through many discussions, I have received a variety of opinions.

Thank you very much.

Bibliography

[1] Mohammad Hesam Hesamian, Wenjing Jia, Xiangjian He, and Paul Kennedy. Deep learning techniques for medical image segmentation: achievements and challenges. *Journal of digital imaging*, 32(4):582–596, 2019.

[2] Tongxue Zhou, Su Ruan, and Stéphane Canu. A review: Deep learning for medical image segmentation using multi-modality fusion. *Array*, 3:100004, 2019.

[3] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[5] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016.

[6] Yu Gordienko, Peng Gang, Jiang Hui, Wei Zeng, Yu Kochura, Oleg Alienin, Oleksandr Rokovyi, and Sergii Stirenko. Deep learning with lung segmentation and bone shadow exclusion techniques for chest x-ray analysis of lung cancer. In *International conference on computer science, engineering and education applications*, pages 638–647. Springer, 2018.

[7] Sheikh Bilal Ahmed, Syed Farooq Ali, Jameel Ahmad, Muhammad Adnan, and Muhammad Moazam Fraz. On the frontiers of pose invariant face recognition: a review. *Artificial Intelligence Review*, 53(4):2571–2634, 2020.

[8] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1875–1882, 2014.

[9] Yuyan Liu, Xiaoying Gong, Jiaxuan Chen, Shuang Chen, and Yang Yang. Rotation-invariant siamese network for low-altitude remote-sensing image registration. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:5746–5758, 2020.

[10] Michiaki Ueda, Keijiro Kanda, Junichi Miyao, Shogo Miyamoto, Yukiko Nakano, and Takio Kurita. Invariant feature extraction for cnn classifier by using gradient reversal layer. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 851–856. IEEE, 2021.

[11] Randall Balestriero and Yann LeCun. Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. *arXiv preprint arXiv:2205.11508*, 2022.

[12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[13] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[14] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

[15] Murugan Ramyaa, Mojoo Jonathan, and Takio Kurita. Supervised learning for convolutional neural network with barlow twins. In *ICANN2022 (submitted)*, 2022.

[16] P. Radau, Y. Lu, K. Connelly, G. Paul, A. Dick, and G. Wright. Evaluation framework for algorithms segmenting short axis cardiac mri. *The MIDAS Journal-Cardiac MR Left Ventricle Segmentation Challenge*, 07 2009.

[17] A. Emre Kavur, N. Sinem Gezer, Mustafa Barış, Sinem Aslan, Pierre-Henri Conze, Vladimir Groza, Duc Duy Pham, Soumick Chatterjee, Philipp Ernst, Savaş Özkan, Bora Baydar, Dmitry Lachinov, Shuo Han, Josef Pauli, Fabian Isensee, Matthias Perkonigg, Rachana Sathish, Ronnie Rajan, Debdoot Sheet, Gurbandurdy Dovletov, Oliver Speck, Andreas Nürnberger, Klaus H. Maier-Hein, Gözde Bozdağı Akar, Gözde Ünal, Oğuz Dicle, and M. Alper Selver. CHAOS Challenge - combined (CT-MR) healthy abdominal organ segmentation. *Medical Image Analysis*, 69:101950, April 2021.

[18] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[19] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on*

International Conference on Machine Learning, ICML'10, page 807–814, Madison, WI, USA, 2010. Omnipress.

[20] John Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989.

[21] Shervin Minaee, Yuri Y. Boykov, Fatih Porikli, Antonio J Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

[22] Nima Tajbakhsh, Laura Jeyaseelan, Qian Li, Jeffrey N. Chiang, Zhihao Wu, and Xiaowei Ding. Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. *Medical Image Analysis*, 63:101693, 2020.

[23] Dzung L. Pham, Chenyang Xu, and Jerry L. Prince. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2(1):315–337, 2000. PMID: 11701515.

[24] Fausto Milletari, Seyed-Ahmad Ahmadi, Christine Kroll, Annika Plate, Verena Rozanski, Juliana Maiostre, Johannes Levin, Olaf Dietrich, Birgit Ertl-Wagner, Kai Bötzel, et al. Hough-cnn: deep learning for segmentation of deep brain regions in mri and ultrasound. *Computer Vision and Image Understanding*, 164:92–102, 2017.

[25] Jens Kleesiek, Gregor Urban, Alexander Hubert, Daniel Schwarz, Klaus Maier-Hein, Martin Bendszus, and Armin Biller. Deep mri brain extraction: A 3d convolutional neural network for skull stripping. *NeuroImage*, 129:460–469, 2016.

[26] Yuyin Zhou, Lingxi Xie, Wei Shen, Yan Wang, Elliot K Fishman, and Alan L Yuille. A fixed-point model for pancreas segmentation in abdominal ct scans. In *International conference on medical image computing and computer-assisted intervention*, pages 693–701. Springer, 2017.

[27] Novanto Yudistira, Muthusubash Kavitha, Takeshi Itabashi, Atsuko H Iwane, and Takio Kurita. Prediction of sequential organelles localization under imbalance using a balanced deep u-net. *Scientific reports*, 10(1):1–11, 2020.

[28] Lukman Hakim, Muthu Subash Kavitha, Novanto Yudistira, and Takio Kurita. Regularizer based on euler characteristic for retinal blood vessel segmentation. *Pattern Recognition Letters*, 149:83–90, 2021.

[29] Lukman Hakim, Huipeng Zheng, and Takio Kurita. Improvement for single image super-resolution and image segmentation by graph laplacian regularizer

based on differences of neighboring pixels. *Manuscript submitted for publication*, 2021.

[30] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555*, 2022.

[31] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.