CHAPTER 7

TRANSFER LEARNING AND MODIFIED UNET BASED SEGMENTATION METHOD ON NORMAL AND NOISY CARDIAC MR IMAGES

Highlights of the Chapter

- *Transfer learning and modified U-Net-based multiclass segmentation of cardiac MR images is done.*
- *Transfer learning method gave competitive results compared to state-of-the-art methods with lesser training time and resource requirement.*
- *U-Net modification with noise stifler block and SeLU as activation function improved the segmentation accuracy even in the noisy CMRI along with faster convergence of the learning curve.*

Abstract

The segmentation of cardiac MR images requires extensive attention as it needs a high level of care and analysis for the diagnosis of affected part. The advent of deep learning technology has paved the way for efficient, automatic and reliable segmentation of medical images for proper diagnosis. This manuscript proposes an augmented end to end trainable deep learning architecture for efficient segmentation of various domains of medical images. The model incorporates the depth wise separable convolution and group normalization as basic building blocks. Moreover, a noise stifler block is also induced between the encoder and decoder to counter the noise in the medical images. This path helps in delineating the precise boundary contours as the noise often reduces the

boundary segmentation capability of the segmentation network. The network trained once produces exceedingly good results for the images of other datasets. An improvement of above (5 ± 0.03) % and (3.5 ± 0.02) % was observed in the Jaccard index and Dice score for cardiac MR images. The results are statistically validated as $p < 0.05$. The automatic computer investigated approach can help in reducing the burden on the medical system by producing accurate and reliable results. The algorithmic results were clinically verified by the senior radiologists by comparison with the manually segmented images. The training time of the network was about 30% less than U-net.

7.1. Introduction

Image segmentation is a process that subdivides an image into its constituent parts or region of interest. The problem under analysis determines the level of subdivision in the image. It is a low-level operation as it operates pixel-wise. A corresponding category label is provided to each pixel in the image by the image segmentation operation[1] [2]. The basic requirement for medical image segmentation is to study the changes in pathological and anatomical structures for analysis of the problem. The accurate and precise extraction of features forms the area being segmented is reasonably necessary for the medical experts to provide the correct diagnosis [3]. The primary medical image segmentation task includes segmentation of brain tumour, skin cancer tissues segregation, cardiac image segmentation, analysis of ventricle cavities by image segmentation, outlining of liver tumor, optical disc segmentation, pulmonary nodules, cell segmentation etc. The advancement in medical imaging technology has paved the way for efficient analysis of the disease affected regions. X-Ray [4], Computed Tomography (CT) [5], MRI [6] and Ultrasound [7] are the main modalities in the medical imaging domain, which are widely used for the detection and

analysis of diseases. Thus, extreme attention is given to the images of these modalities for segmentation and other imaging domain applications.

The primitive approaches for the medical image segmentation were dependent on active contour matching, statistical models, edge locations template matching. Basically, image segmentation is categorized into manual, semi-automatic, and fully automatic segmentation. Manual segmentation requires the experience and expertise of radiologists and doctors to draw a precise region of interest (ROI) which is time-consuming, often impractical for a large dataset, and involves significant variabilities as well. The semi-automatic methods require handcrafted features as well as manual interference for the processing steps [8]. The fully automatic methods which are gaining importance has to lead the way for the computer-based approaches to fully take over the feature extraction as well as segmentation process. With the advent in the field of deep learning, fully automatic methods are producing reliable and accurate results [9]. These methods are self-capable in feature extraction and learning. Deep learning-based architectures use supervised learning techniques for the processing of images. The revolutionary work in the field of medical image segmentation was presented by Olaf Ronneberger in the form of U-net [10]. The end-to-end trainable U-net architecture is capable of producing very accurate results while getting trained on very few images. The advantage of U-net is that input uses data patches which solves the problem of fewer data as the patches are equivalent to the data augmentation. Moreover, the target category can be located by the output results. Other end to end trainable image segmenting network includes Fully Convolutional Networks (FCN) [11] [12], Deeplab [13], SegNet [14] etc. The pixel-level segmentation was achieved using the SegNet model as it incorporates the symmetric structured encoder-decoder network, which is based on the semantic segmentation concept of FCN. The FCN deploys locally connected layers and avoids the use of dense layers. The down sampling path in

FCN extracts the features while the up-sampling path localizes the object. The deep lab was developed by Google as an open-source model for image segmentation. It up samples the output of the last convolutional layer and evaluates pixel-wise loss. The FCN was used by Zhou et al. for the segmentation of nineteen organs in CT images [15]. A cascaded FCN was proposed by Christ et al. [16], which was incorporated using the superposition of a series of FCN. In this approach, context features were extracted from the prediction map. Automatic segmentation of brain tumor using CCN-PR-Segnet was proposed by Tripathi et al. for efficient segmentation of brain tumor images [6]. The adversarial network proposed by Goodfellow et al. [17] were used by Luc et al. [18] to segment the images by training a CNN based segmentation network and an adversarial network. Segmentation Adversarial Network (SegAN) proposed by Xue et al. [19] used the U-net model as the generator for GAN. This network is subdivided into segmentation network and critic network. These two are trained alternately to finally produce a good segmentation model. Zhang et al reviewed Generative Adversarial Network based fast MRI methods on various anatomical datasets to show its generalizability and robustness for this kind of fast MRI [20]. Wu et al proposed use of multi-channel data for mapping by fusion with attention module and shape information-based post-processing to achieve accurate delineation of both epicardial and endocardial contours [21]. Jin et al proposed using bicubic interpolation for MRI pre-processing considering the low-frequency components of prostate gland and trained an improved 3D V-net (dubbed 3D PBV-Net) for segmentation task. The proposed method effectively improved accuracy of the segmentation of prostate [22].

A high-level 3D semantics capturing network names as Projective Adversarial Network (PAN) was proposed by Khosravan et al [23]. Zhou et al developed a method to systematically asses the size of ventricle of elderly patients for both CT and MRI images and even for images with different

slice thickness and produces superior segmentation results [24]. Pedro et al developed a fully automated post processing framework for in vivo diffusion tensor CMRI using deep learning. A U-Net based network is proposed here to segment the images and results showed a good correlation between segmentation obtained by experienced user and one obtained by proposed method [25]. PAN uses a segmentation and two adversarial networks as the basic constituents of the architecture. 2D projection is used to integrate high-level 3D information without increasing the complexity in the segmentation process. Deep contour network (DCAN) [26] proposed by Hao Chen et al. generates the predictions of both objects and contours for the segmentation process. Peter Naylor et al. [27] developed a method by replacing binary annotation with a distance map to train the network to predict pixel by pixel segmentation. Zhang et al[28] proposed a model to reduce the difficulty of feature extraction and proposed a new "Categorical Dice" loss. Also, Zhou et al. [29] developed a retrospective study to segment Ventricle images in elderly patients and showed to produce excellent results. Chen et al proposed inter-cascade GAN for segmentation task of unbalanced atrial targets from late gadolinium-enhanced cardiac magnetic resonance images. It first investigates an adaptive attention cascade to automatically correlate segmentation of unbalanced tasks and then applies adversarial regularization to segment the scars [29]. Multiview recurrent aggregation network (MV-RAN) proposed by M. Li et al. utilized a unified framework for segmentation and classification of echocardiographic sequences segmentation [30]. A modified 2 stage U-net proposed by M. Pop et al. simultaneously learned to detect ROI in the full volume and classified voxels without compromising the original resolution. This approach achieved better segmentation performance as compared to state of the art methods [31]. A. F. Frangi et al. proposed a probabilistic deep voxel-wise dilated residual network for segmentation of whole heart in 3D MR images [32]. An iterative algorithm for segmentation of left ventricle (LV) was proposed by

Yuanhan Mo. Fusion of deep learning network with the novel dynamic-based labeling scheme was utilized in this approach [33]. G. Yang, J. Chen, Z. Gao et al. proposed a joint segmentation approach which utilizes multiview two-task (MVTT) recursive attention model. This method is used to segment the left atrium (LA) in 3D CMR images and in addition it also delineates the scar on the same dataset [34]. For segmentation of the peripheral zone (PZ) and transition zone (TZ) of the prostate with uncertainty estimation Liu et al. proposed spatial attentive Bayesian deep learning network. The overall uncertainty was found to be the highest with the actual model performance [35]. A fully CNN with a novel feature pyramid attention mechanism was developed by Y. Liu et al. for automatic segmentation of prostate zones. This automatic segmentation method achieved highest accuracy in segmentation of PZ and TZ [36].

The presented research's motivation is to develop an efficient deep learning-based architecture to segment the noisy MR images for reliable analysis of the disease [23]. The primary source of noise in MR images is thermal in nature, which originates from the stochastic motion of free electrons. Thermal noise is a white additive and follows Gaussian distribution having variance (σ) and zero mean (μ). The raw data for MR images is K-space data which is corrupted by Gaussian noise. The noise distribution of data changes when the magnitude of data is computed as this process is non-linear. The Gaussian noise distribution of data is converted to Rician after this process. This noise creates problems in the proper segmentation of the boundary of the objects. The main objective of this work is to effectively segment the MR images corrupted with the Rician noise as the real-time MR images contain some percentage of unwanted Rician distributed noise. The proposed is integrates noise stifler block for noise suppression at the time of segmentation. The accurate diagnosis of the disease is dependent on the perfection in interpretability of the image. Thus, precise and accurate segmentation is quite helpful for the medical practitioner to plan the

diagnosis. The proposed model produces more and accurate results in comparison to its conventional counterpart.

7.2 Method

7.2.1 Transfer learning Algorithm

Recently, several methods have been proposed to tackle the problem of developing CNN-based segmentation networks. A fully Convolutional Network (FCN) [35] is one of the most successful methods which gives entire image segmentation in a single forward pass. FCN based methods allow training networks in the end-to-end manner of input images of any size. FCN is further improved to obtain better segmentation results known as Feature Pyramid Network (FPN) and U-Net.

7.2.1.1 Feature Pyramid Network [36] [37]

FPN takes leverage of convolutional network pyramidal feature hierarchy where networks has semantics from low to high levels and builds a feature pyramid with high-level semantics throughout. It takes images of same scale of any size as input and outputs feature maps of similar size. The process is independent of backbone architecture and can take several networks as encoder. FPN consists of two pathways: bottom-up and top-down. The bottom-up pathway is a feed-forward convolutional network used for feature extraction at several scales with a scaling step of 2. These convoluted pathways can be any CNN [38] like ResNet, DenseNet, InceptionNet [39] , VGG. The general architecture of the FPN based network is shown in figure 7.1. In the bottom-up pathway, as one moves up, the spatial resolution decreases while high-level structure features extracted increase. At the top of the bottom-up pathway, 1x1 convolution is used to reduce the channel depth. Then two 3x3 convolution were applied which gives the first feature map for segmentation. In the top-down pathways, as one goes from the top down, the previous layer is up

sampled by 2 using nearest neighbors up sampling. Again, 1x1 convolution is applied to the corresponding feature maps in the bottom-up pathway and then added element-wise. Two 3x3 convolutions are then applied to output the final feature map for image segmentation. In the end, all feature maps having 128 channels are concatenated resulting in 512 channels. A 512 3x3 convolution filter is then applied with batch normalization and RELU activation and then a 1x1 convolution is applied to obtain a final feature map.

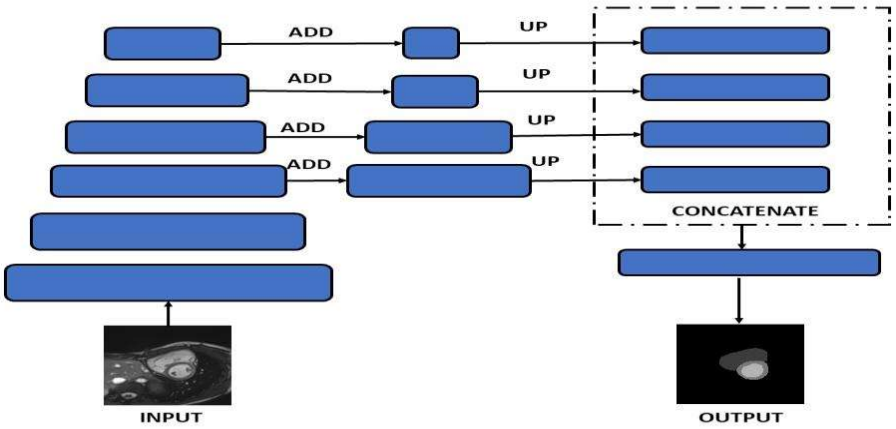

Figure 7.1: **-** Feature Pyramid Network generalized architecture with bottom-up and top-down pathways

7.1.1.2 U-Net

U-Net is the most commonly used network for all segmentation purposes. It was developed by Olaf Ronneberger [19] for biomedical image segmentation as it can be trained on a fewer number of images. It is an end-to-end trainable network. It is successive modification to autoencoders where the pool operation is switched with up-sampling operation to supplement the contracting network layers. It consists of a contracting path and an expansive path. The contracting path consists of two 3x3 convolutions followed by a rectified linear unit (RELU) [40] and a 2x2 max pooling [41] operation with stride 2 for down sampling. The number of features channel is doubled after each down sampling. At every step of expansive path, up sampling of the feature map is carried followed by a 2x2 convolution to half the number of feature channels and

concatenated with corresponding cropped feature map of contraction path. The concatenation is followed with two 3x3 convolutional layers, each with RELU activation. It is required to crop because the border pixels were lost at every convolution. At the final output layer, a 1x1 convolution is used to map each 64-component feature map to desired class. A total of 23 convolutional layer are there in the network. The use of dense layers is avoided so that it can accept images of any size.

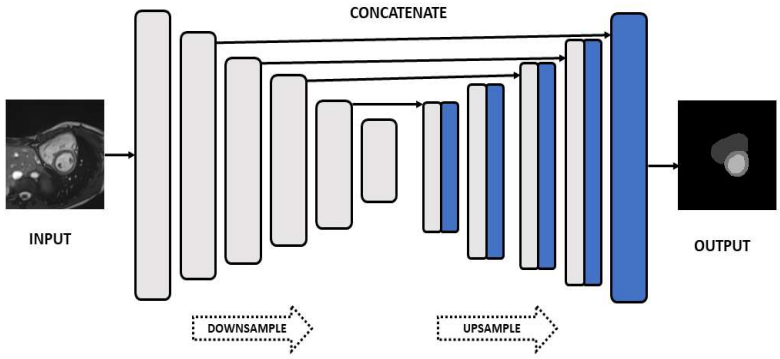

Figure 7.2: - U-Net generalized architecture with contractive and expansive paths

Segmentation model GitHub repository for PyTorch (Python) is a very handy and efficient library for image segmentation tasks. A total of 8 model architectures including but not limited to U-Net, FPN and PSPNet are available for binary and multiclass image segmentation with 99 encoders pre-trained on ImageNet dataset. Pre-trained models provide a better and faster convergence compared to training a model from scratch and thus resources required are considerably reduced. In the project, they placed the well-established networks like DenseNet, ResNet, and Vgg in place of the encoder in the case of U-Net and in place of the bottom-up pathway of FPN. Models are then trained on a 2012 ILSVRC ImageNet dataset. Apart from models, the repository also provided the preprocessing of the input images according to the network requirement and several losses and evaluation metrics are available to choose and optimize the particular work to be done.

7.2.2 Modified U-Net Algorithm

The proposed network comprises depth wise separable convolution, group normalization, Scaled exponential Linear Unit (SELU) and noise stifler block. The depth wise separable convolution [37] with bottleneck connections [38] [40] is used in this end to end trainable segmentation network, as shown in figure 7.2. The in-between inputs are encoded by the bottleneck connections. The transformation from lower-level units such as pixels to higher-level illustration such as image is performed by the middle layers. It consists of three convolution layers in the block. The layers are named as 1) Expansion Layer, 2) Depth wise convolutional Layer and, 3) Projection Layer. For each input channel, a dedicated single filter is incorporated. A pointwise 1×1 convolution is used for the process of combining the outputs of depth wise separable convolution. Filtering and combining is collectively termed as depth wise separable convolution. This process is split into two parts, a different layer for filtering and a different layer for combining, as shown in figure 7.3(a) when depth wise convolution is incorporated. Figure 7.3(b) shows the overall architecture of the network.

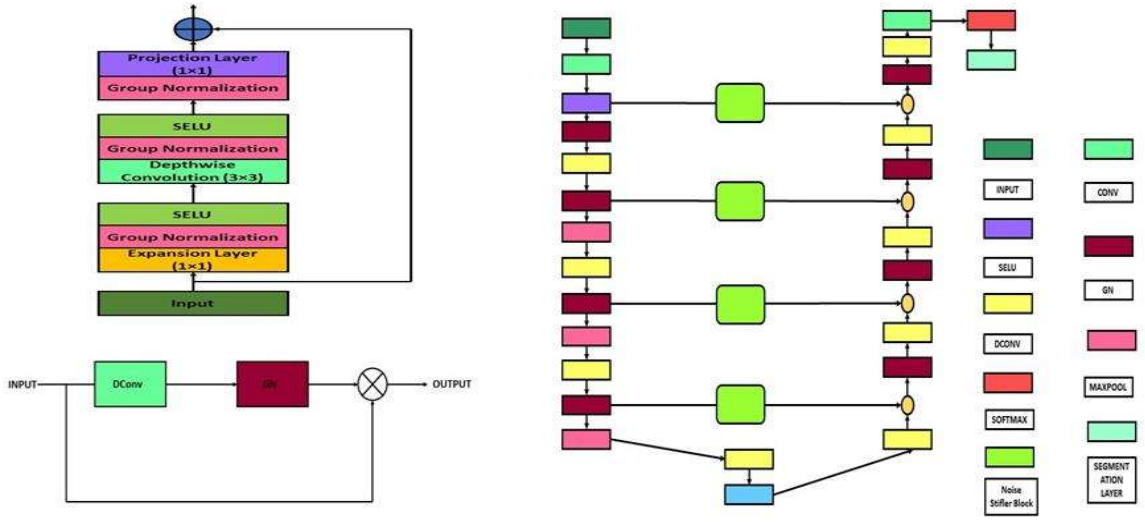

Figure 7.3 (a) Depth wise Separable block (b) Architecture of the proposed network (c) Noise stifler Block

Depth wise convolution with one filter per input channel (input depth) can be written as:

$$\hat{G}_{k.l,m} = \hat{K}_{i,j,m}.F_{k+i-1,l+j-1,m} \qquad (1)$$

Where is the depth wise convolutional kernel of size $D_K \times D_K \times M$ where the m^{th} filter is applied to the m^{th} channel to produce the m^{th} channel of the filtered output feature map. The following equation gives the cost of convolution for depth wise separable convolution:

$$D_K.\ D_K.M.\ D_{F.}\ D_F + M.N.\ D_{F.}\ D_F \qquad (2)$$

Equation 2 is the sum of depth wise convolution and 1×1 pointwise convolution. This 1×1 pointwise convolution layer causes the number of channels to be lower and is also known as the Projection Layer and Bottleneck. The projection layer projects the data with higher dimensions into a tensor with a lower number of dimensions. The first layer in figure 7.1 is named the expansion layer, which possesses the opposite characteristics of the projection layer. It has a smaller number of input channels than the output channels. The hyperparameter expansion factor determines how much the data will be expanded. Each layer incorporates group normalization and Scaled exponential Linear Unit (SELU) [40] as activation function. The use of non-linearity may destroy meaningful information, so the use of activation function is avoided in the projection layer. The overall cost of the computation is reduced by the incorporation of depth wise separable convolution because of the reduction in the number of multiplications. The training parameters are also less as compared to standard convolution. Thus overfitting issues are reduced to a large extent. The image is transformed once, so the computation complexity is reduced. The boundary and object-related information are precisely captured in the case of depth wise separable convolution. This is because, in the case of regular convolution, an image is transformed m times where m denotes the number of channels, while in the case of depth wise separable convolution, the image is transformed only once and elongated to m channels. The reduction in the number of

multiplications preserves the boundary details, as with each additional multiplication, the boundary details are lost. Scaled exponential unit (SELU) is used instead of RELU as the former shows robust performance than the latter. The SELU cannot die because of the negative slope, and thus it removes the problem of vanishing gradient in deep neural networks. It possesses the self-normalization property. SELU is a trade-off between RELU and Leaky RELU as the constant slope of the leaky RELU may elevate the risk of vanishing gradient. The SELU is given by the following equation:

$$f\ (\alpha, x) = \tau \begin{cases} \alpha e^{x} & for\ x < 0 \\ x & for\ x \geq 0 \end{cases} \qquad (3)$$

For normalizing the data, group normalization [41] is used, which normalizes the data across the groups. The channels are divided into groups before the normalization. The accuracy is stable over a wide batch size in case of group normalization as there is no dependence on batch size. Thus, the main drawback of handling the large batch size of batch normalization is eliminated by the use of group normalization. The process of group normalization requires the computation of the mean and variance of the inputs over the specified groups of channels. The following equation describes group normalization:

$$\widehat{a_i} = \frac{a_i - \mu_g}{\sqrt{\sigma_g^2 + \varepsilon}} \qquad (4)$$

Where a_i is the input, μ_g is the mean and is the variance. ε is the property to improve numerical stability for the slight group variance. The max-pooling operation is performed in the subsequent stage with window size 2 × 2 and stride 2. The related contents of the features are rejected by this layer, and the features with vital information are stored. Translational invariance is achieved over a small spatial shift using this max-pooling operator. The SoftMax operator is applied next to the input data, which turns the raw score, i.e. logits, into probability which sums to 1. It acts as a

multiclass sigmoid function. The SoftMax function is used for the determination of probability computed once of multiclass. The following equation represents the SoftMax function:

$$Y_r(x) = \frac{\exp(a_r(x))}{\sum_{j=1}^{k} \exp\left(a_j(x)\right)} \tag{5}$$

Where r signifies the class $0 \leq Yr \leq 1$, and the inputs and outputs of the layer are denoted by x and Y, respectively. The segmented result generated by the SoftMax [42] corresponds to the class with the maximum probability for each pixel in the image. Finally, a categorical label is provided to every pixel in the image by the pixel classification layer. For optimisation of network parameters, the cross-entropy loss [43] function is used in the network. The loss was calculated using the following equation:

$$loss = -\frac{1}{N}\sum_{i=1}^{M} T_i \log(X_i) \tag{6}$$

Where Xi and Ti are the response of network and target value, the total number of responses in the image is denoted by M, and N is the total number of responses in X. The noise stifler block is inserted between the connections from the encoder and decoder. The objective of this block is to suppress the noise from the image features. By reducing the noise, the spurious signal may be eliminated from the image feature, which is the unwanted property of the real-time medical images. The noise stifler block is shown in figure 7.4. This block performs pointwise multiplication of group normalized depth wise convolved features with the original features. The pointwise multiplication reduces the high frequency noise components. Thus, the noise in the feature maps gets reduced which is quiet desirable for effective segmentation of MR images.

7.3 Experiments

7.3.1 Transfer learning Algorithm

In this study, the dataset used is the Automatic Cardiac Diagnosis Challenge (ACDC) dataset. Each CMR image data was acquired with different MRI machines on different subjects by different

technicians. This led to inconsistency in CMR images in the datasets. For each image, pixel spacing, pixel size, image size, contrast variation is different and so considering these factors, pre-processing steps are selected. Datasets are first preprocessed with ROI cropping, Contrast limited adaptive histogram equalization (CLAHE) [42], and pixel resizing. Dataset is then split into 60:20:20 ratio for training, validation, and test set. The data pipeline for training models consists of several data augmentation for training sets to increase the number of training images as well as to increase the variation in the training data. Pre-processing particular for the model is then applied and models are trained. These trained models are then used to predict on a test dataset and evaluation metrics like Dice score and Jaccard Coefficient are calculated to check the efficiency of the model to segment the images.

Dataset: - Automatic Cardiac Diagnosis Challenge (ACDC) dataset contains MR images of the Heart in a Short-axis (SAX) view of 100 patients with manual segmentation of LV, Myocardium, and RV for End-Systole and End-Diastole phase in the NIfTI (Neuroimaging Informatics Technology Initiative) image format. CMR image acquisitions were done over 6 years using two different MRI scanners of Siemens (1.5 T Siemens Area, Siemens Medical Solutions, Germany & 3.0 T Siemens Trio Tim, Siemens Medical Solutions, Germany). CMR images are acquired in breadth-hold with SSFP sequence in short-axis orientation. Series of slices cover from base of LV to apex with a thickness of 5 or 8 mm with an interslice gap of 5 mm, spatial resolution varying from 1.37 to 1.68 mm^2/pixel and 28 to 40 images per patient.

Pre-processing: - ACDC dataset is provided with an ROI center and based on that the images were first cropped to 128x128 pixel size. This provided benefits like reduction in background, pixel intensity interference due to other organs, mitigate class imbalance, and accelerated computation due to reduction in input image size. As the images are taken from different scanners

at different sites, there is a wide range of intensity variation. To overcome the effect of intensity variation during training, images were first normalized using min-max normalization. The training set is then affine transformed (rotation by 90°, vertical flipping by 180°, and horizontal flipping by 180°) to augment to overcome the overfitting and increase the model generalization.

Training protocol: - The segmentation network is implemented in Python using Pytorch framework and trained on Nvidia K80 GPU provided by Google Collaboratory. The ADAM optimizer is used with the initial learning rate of 0.001 and is trained for 100 epochs with a batch size of 4. Keras callback method is used to reduce the learning rate on the plateau.

Loss function:- For a set of events, a measure of the difference between the probability distribution is known as cross-entropy [43]. Using cross-entropy as a loss function instead of other loss functions like the sum of squares leads to faster training and better generalization of the model. The loss calculated by cross-entropy is the average per-pixel loss and the per-pixel loss is calculated discreetly. As a result, cross-entropy considers loss at micro sense. Cross entropy is defined as:

$$Loss_{CE} = -\frac{1}{m}\sum [y^i \, log \, p^i + (1 - y^i) \, log(1 - p^{i)}] \quad (1)$$

where m is the number of samples, y^i is the label of the sample, and p^i is the predicted probability value, $p^i \in (0, 1)$. If there is a class imbalance in the training set, training will be difficult. To overcome this, the loss function is conjugated with dice loss which is a statistic developed in the 1940s to gauge the similarity between two samples. It was brought to the computer vision community by Milletari et. al..[44] in 2016 for 3D medical image segmentation. Dice loss [45] is defined as:

$$Dice\ Loss = 1 - \frac{2*\sum_{pixels} y_{true} y_{pred}}{\sum_{pixels} y_{true}^2 + \sum_{pixels} y_{pred}^2} \quad (2)$$

The segmented results were evaluated using well known metrics namely Jaccard Index [46] [47], Dice score [48], Matthews correlation coefficient (MCC) [49], accuracy, specificity, recall and precision.

7.3.2 Modified U-Net Algorithm

The following steps were followed for performing the experiment:

Step 1: Rician noise [44] was added manually in the MR images. The percentages of added noise were 1%, 3%, 5% and 7%. (Rician noise was added because the noise corruption follows Rician distribution in MR images.)

Step 2: The proposed network was implemented using the tensor flow framework on Python 3.7.

Step3: The network was supplied with the original images and the noisy images for segmentation.

Step 4: The results were obtained, and evaluation metrics were calculated.

Step 5: For comparison of results, step 2 to 4 were repeated for other networks.

Step 6: The comparative study of the results of various methods was performed.

Datasets and Evaluation Metrics:

In this paper, two datasets were used for segmentation. The heart image segmentation datasets used in this paper are Sunny Brook Dataset (SCD) and Automatic Cardiac Diagnosis Challenge (ACDC). The Sunnybrook Cardiac MR Left Ventricle Segmentation Challenge (MICCAI 2009) dataset consists of 45 CMR images from patients having four pathological conditions like heart failure with ischemia, heart failure without ischemia, hypertrophic cardiomyopathy and normal subjects. Manually drawn contours for the endocardium and epicardium are provided in text format, which consists of contour points. The automatic Cardiac Diagnosis Challenge (ACDC)

dataset contains MR images of the Heart in a Short-axis (SAX) view of 100 patients with manual segmentation of LV, Myocardium, and RV for End-Systole and End-Diastole phase in the NIfTI (Neuroimaging Informatics Technology Initiative) image format. The network was trained on the ACDC dataset for heart segmentation while tested on 20% part of the same dataset and full SCD dataset. The results obtained were evaluated using the metrics: accuracy, Jaccard Index [45] [46], Dice Score and Matthews correlation coefficient (MCC) [47]. Accuracy is given as the ratio of correctly classified pixels, irrespective of class, to the total number of pixels in the dataset. Jaccard is defined as the fraction of the overlapped area between the predicted segmentation and the ground truth data and the union of the area of segmentation being predicted and ground truth data. The Dice score is defined as the harmonic mean (F1-measure) of the precision and recall values with a distance error tolerance to decide whether a point on the predicted boundary matches the ground truth boundary or not [48]. The Matthews correlation coefficient (MCC) is used to firmly quantitative the analysis of classification as the mathematical properties of MCC handles both dataset imbalance and their invariants effectively. It gives correct predictions for both majorities of the negative cases and positive cases, independently of their ratios in the overall dataset. The following equations describe the above stated metrics:

$$Jaccard\ (U,V) = \frac{U \cap V}{U \cup V} \tag{7}$$

where U is the predicted segmentation data, and V is the ground truth data.

$$Dice\ Score = \frac{2 \times P \times R}{R+P} \tag{8}$$

where, P and R denotes the precision and recall values respectively.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

$$MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{10}$$

where TP, TN, FP and FN are the four cardinalities of confusion metrics.

The proposed Convolutional network was implemented using Keras API in Python 3.6 configured with TensorFlow library as the backend. The python scripts were executed on Python 3 Google compute engine backend provided by Google Collaboratory. It provides a virtual machine that has 2 x Intel(R) Xeon(R) @ 2.3 GHz CPUs, 12 GB RAM and an NVIDIA K80 GPU with 16 GB GRAM. Training of each model is done for 100 epochs with ADAM optimizer, and the learning rate is kept at 0.0001.

7.4 Results

7.4.1 Transfer learning algorithm

Figure 7.4 illustrates segmentation results for the End-Diastolic Phase of the cardiac cycle obtained using the Feature pyramid network architecture. Figure 7.4(a) is the original image while 7.4(b) illustrates the manual segmentation (ground truth) provided by the experts. Figures 7.5(c), 7.5(d), and 7.5(e) are the segmentation predictions obtained from the Feature Pyramid network with bottom-top pathways replaced with DenseNet [20], ResNet [50], and VGG [51] respectively. Mean Dice score obtained for LV, MYO and RV class of segmentation are 0.954, 0.877 and 0.930 for encoder DenseNet trained model while 0.955, 0.868 and 0.935 for ResNet and 0.933, 0.820 and 0.886 for VGG.

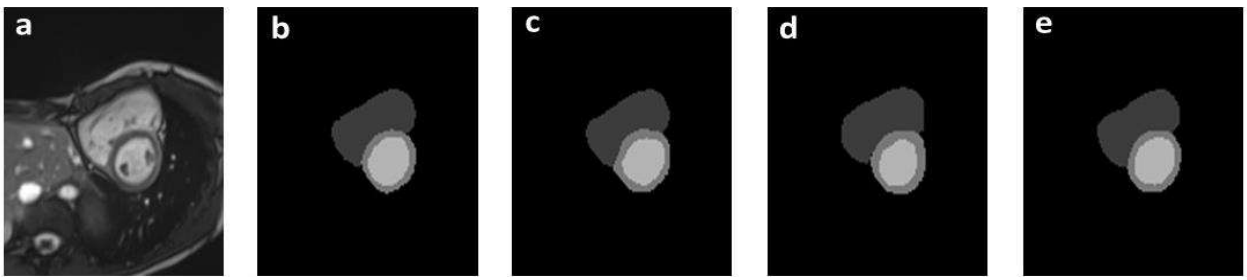

Figure 7.4: - End-diastolic FPN network (a) Original Image (b) Ground Truth (c) FPN with DenseNet encoder Prediction (d) FPN with ResNet encoder Prediction (e) FPN with VGG Prediction

Similarly, Mean Dice score obtained for LV, MYO and RV class of segmentation are 0.956, 0.878 and 0.934 for U-Net with encoder as DenseNet while 0.952, 0.876 and 0.936 for U-Net with encoder ResNet and 0.966, 0.928 and 0.943 for U-Net with encoder VGG. Segmentation results for U-Net-based network architecture are shown in figure 7.5. It can be noted that the dice score for U-Net with VGG encoder model gives better results compared to other architecture-encoder configurations. Jaccard index for all networks is shown in Tables 7.2 & 7.3 respectively. Results are further validated by evaluating metrics like Precision, Specificity, Sensitivity, and Matthews's correlation coefficient (MCC) shown in figure 7.5.

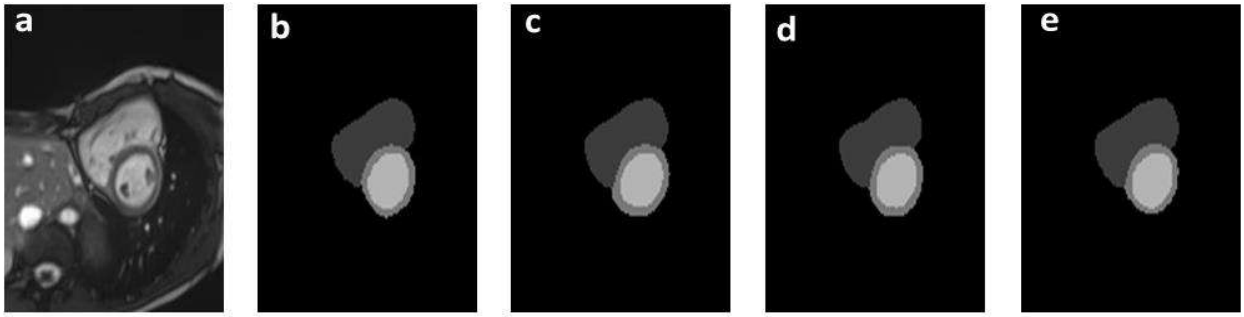

Figure 7.5: - End-diastolic U-NET network (a) Original Image (b) Ground Truth (c) U-NET DenseNet Prediction (d) UNET with ResNet encoder Prediction (e) U-NET with encoder VGG Prediction

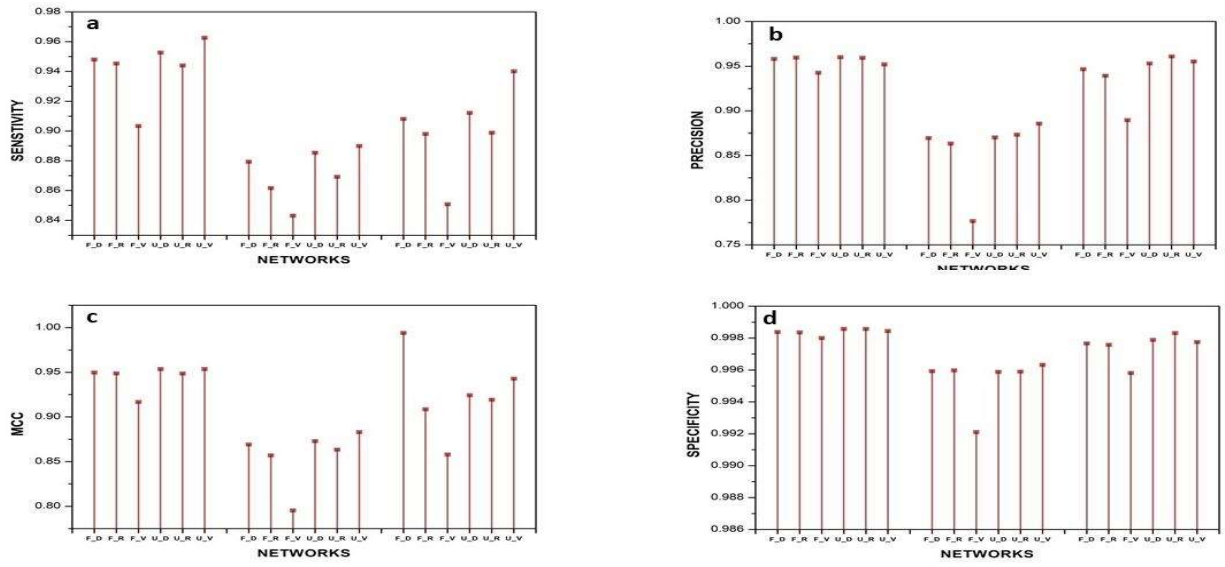

Figure 7.6: - Evaluation metrics for different networks for End-Diastolic Phase (a) Sensitivity (b) Precision (c) MCC (d) Specificity

#F_D: FPN with DenseNet encoder, F_R: FPN with ResNet encoder, F_V: FPN with VGG encoder, U_D: U-Net with DenseNet encoder, U_R: U-Net with ResNet encoder, U_V: U-Net with VGG encoder

Figure 7.6 & Figure 7.7 illustrates the segmentation results obtained from FPN and U-Net architectures-based models for End-systolic phase of cardiac cycle. Mean dice score for end-systolic phase for FPN architecture with encoder DenseNet, ResNet, and VGG for LV class are 0.928, 0.929 and 0.896 respectively, and for MYO class are 0.889, 0.886, and 0.844 respectively while 0.907, 0.916 and 0.870 respectively for RV class.

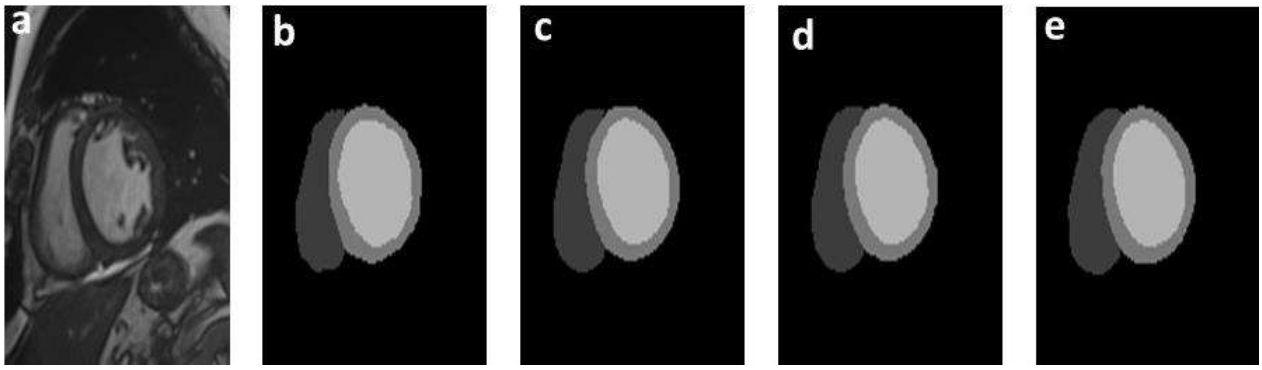

Figure 7.7: - End-Systolic FPN network (a) Original Image (b) Ground Truth (c) FPN with DenseNet encoder Prediction (d) FPN with ResNet encoder Prediction (e) FPN with VGG encoder Prediction

Similarly, Mean dice score for end-systolic phase for U-Net architecture with encoder DenseNet, ResNet, and VGG for LV class are 0.929, 0.923, and 0.936 respectively and for MYO class are 0.888, 0.881, and 0.906 respectively while 0.923, 0.908, and 0.930 respectively for RV class. Jaccard index for the different models is shown in Tables 7.2. Additional validation of segmentation quality is confirmed using metrics like precision, sensitivity, specificity, and MCC are shown in figure 7.8. Results show that the model of U-Net architecture with vgg19 as encoder gives the best results among different configurations.

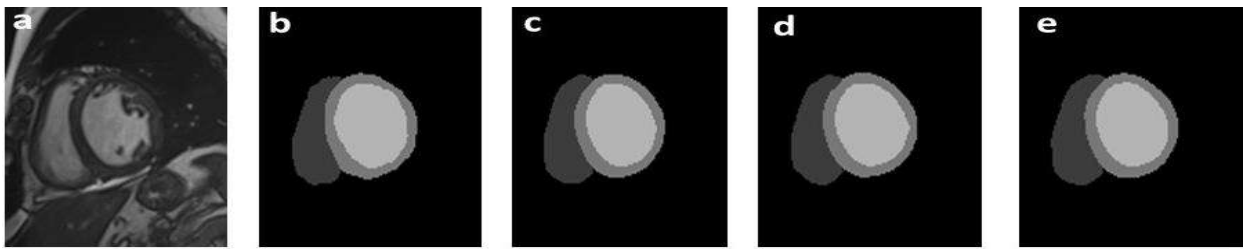

Figure 7.8: - End-Systolic U-NET network (a) Original Image (b) Ground Truth (c) UNET with DenseNet encoder Prediction (d) U-NET with ResNet encoder Prediction (e) U-NET with VGG encoder Prediction

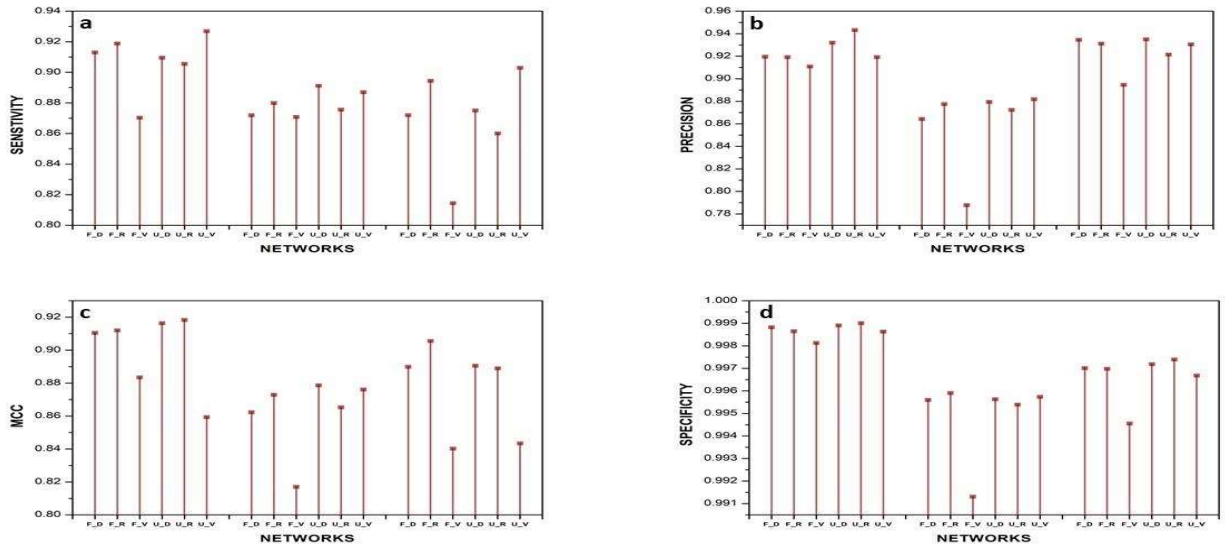

Figure 7.9 Evaluation metrics for different networks for End-systolic Phase(a) Sensitivity (b) Precision (c) MCC (d) Specificity

#F_D: FPN with DenseNet encoder, F_R: FPN with ResNet encoder, F_V: FPN with VGG encoder, U_D: U-Net with DenseNet encoder, U_R: U-Net with ResNet encoder, U_V: U-Net with VGG encoder

Table 7.1: - Dice Score for End-Diastolic and End-Systolic cardiac phase for different models

F1 Score (Dice Score)						
Model (Architecture/ Encoder)	**End Diastolic**			**End Systolic**		
	LV	**MYO**	**RV**	**LV**	**MYO**	**RV**
FPN/DenseNet	0.954	0.877	0.930	0.928	0.889	0.907
FPN/ResNet	0.955	0.868	0.935	0.929	0.886	0.916
FPN/VGG	0.933	0.820	0.886	0.896	0.844	0.871
U-Net/DenseNet	0.956	0.878	0.934	0.929	0.888	0.923
U-Net/ResNet	0.9512	0.876	0.935	0.923	0.881	0.921
U-Net/VGG	**0.969**	**0.928**	**0.943**	**0.933**	**0.906**	**0.930**

Table 7.2: - Jaccard Index for End-Diastolic and End-Systolic cardiac phase for different models

Mean IOU (Jaccard)		
Model	**End Diastolic**	**End Systolic**

(Architecture/ Encoder)	LV	MYO	RV	LV	MYO	RV
FPN/DenseNet	0.909	0.776	0.869	0.849	0.781	0.821
FPN/ResNet	0.907	0.762	0.856	0.847	0.789	0.840
FPN/VGG	0.861	0.676	0.782	0.804	0.710	0.751
U-Net/DenseNet	0.916	0.783	0.873	0.847	0.797	0.825
U-Net/ResNet	0.908	0.771	0.868	0.855	0.779	0.817
U-Net/VGG	**0.928**	**0.808**	**0.910**	**0.869**	**0.815**	**0.853**

To verify the segmentation accuracy of the models, we compared it to five models (2D U-Net +3D U-Net ensemble by Isensee et. al. [23], Multi atlas-based method combined with 3D CNN for registration by Rohe et. al. [52], 2D U-Net with a Dice loss by Patravali et. al. [53], U-Net with cross-entropy loss and improved dice loss and modified U-Net with Residual block by Yang et. al. [54] proposed earlier and have been stated to be accurate for segmentation of CMR images. Table 7.3 shows dice scores obtained using different methods. The dice score for our method is 2nd in ranking among the above-stated methods with being 1% less than the method proposed by Isensee et. al.. [23].

Table 7.3: - Comparison of overall mean of dice score of three classes for different methods

Dice Score						
Model (Architecture/ Encoder)	End Diastolic			End Systolic		
	LV	MYO	RV	LV	MYO	RV
Isensee et. al.	0.968	**0.946**	0.902	0.931	0.899	0.919
Baumgartner et. al.	0.963	0.932	0.892	0.911	0.883	0.901
Jang et. al.	0.959	0.929	0.875	0.921	0.885	0.895
Proposed	**0.969**	0.928	**0.943**	**0.933**	**0.906**	**0.930**

Hausdorff distance calculated for the different class for segmentation results obtained by Isensee et. al. and our method is shown in table 7.4. It can be concluded that Hausdorff distance for the segmentation results for our methods reduced considerably. Accuracy and Dice score sometimes misleads the analysis of the results so the Matthews correlation coefficient (MCC) is used for better understanding as it takes into account all the four metrics of the confusion matrix (TP, TN, FP, and FN). The Matthews correlation coefficient (MCC) is more mathematically balanced and produces a high score when all the four metrics of confusion metrics give optimum values [55] [56]. The difference in Dice score is mainly due to distortion in the segmentation result at the apical slices. Therefore, compared to other methods, our method performance is superior except for Isensee et. al. [23] to which dice score is comparable and Hausdorff distance is much better. Also, the resources and time required by the proposed method are much less than others. This added benefit of the models makes it more accurate for cardiac MR image segmentation. The statistical paired t-test was performed to validate the statistical significance of the proposed method. The null hypothesis was rejected as the value of p [57] was less than 0.05 (equal to 0.0313) and a confidence level of 95% was achieved.

Table 7.4: - Comparison of mean Hausdorff distance for different classes

Hausdorff						
Model	**End Diastolic**			**End Systolic**		
	LV	**MYO**	**RV**	**LV**	**MYO**	**RV**
Isensee et. al.	7.4	10.1	8.7	6.9	12.2	8.7
Baumgartner et. al.	6.5	12.7	8.7	9.2	14.7	10.6
Jang et. al.	7.7	12.9	9.9	7.1	11.8	8.9
Proposed	**1.71**	**2.29**	**1.96**	**1.63**	**2.21**	**1.80**

7.4.2 Modified U-Net Algorithm

The results obtained from the experiment are presented in this section. The results are divided in two parts:

Results obtained from heart segmentation ACDC dataset:

The segmentation results obtained when the network was trained and tested on the ACDC dataset are shown in figure 7.10. The 60% data was used for training, 20% data was used for validation and rest 20% was used for testing. Table 7.5 shows the comparison of evaluation metrics for the various networks in the study, along with the proposed network.

Results obtained from heart segmentation SCD dataset:

The trained network was tested on the SCD dataset, and the obtained results are shown in figure 7.11. The comparative study of evaluation metrics is presented in Table 7.6. The comparison of sensitivity, specificity and precision is shown in figure 7.7 for the SCD dataset. Table 7.7 shows comparison results from different literature with proposed method for CMR images without noise.

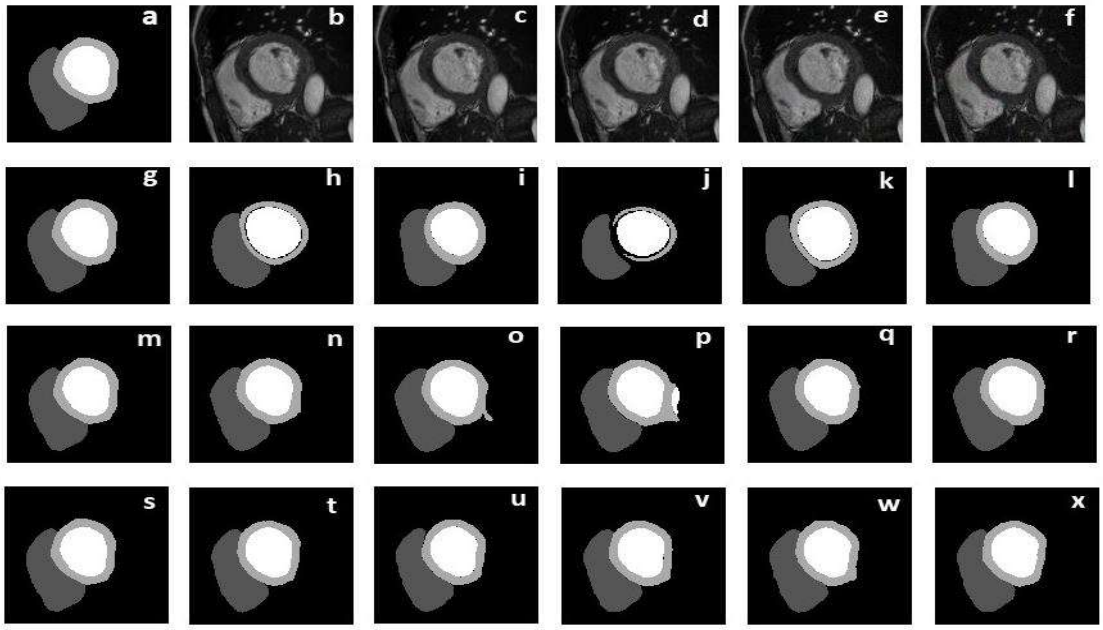

Figure 7.10: Segmentation results of ACDC dataset. First column (a)-(s) presents the mask of the image being segmented, first row (b)-(f) presents the original image and noisy version (1%,3%,5% and 7%) of the original image. Second, third and fourth rows presents the segmented results for Seg-Net (h)-(l), U-net (n)-(r) and proposed network (t)-(x). Second, third, fourth, fifth and sixth column

presents the segmented results for original (h)-(t), 1% (i)-(u), 3% (j)-(v), 5% (k)-(w) and 7%(l)-(x) noise corrupted images.

Table 7.5 Evaluation Metrics Comparison for ACDC dataset

Evaluation Metric	**Images with % noise**															
		Original Image			**1%**			**3%**			**5%**			**7%**		
		LV	MYO	RV	LV	MYO	RV	LV	MYO	RV	LV	MYO	RV	LV	MYO	RV
Jaccard	**Segnet**	0.610	0.390	0.448	0.668	0.531	0.491	0.478	0.199	0.208	0.620	0.429	0.386	0.354	0.459	0.436
	Unet	0.829	0.776	0.745	0.857	0.781	0.765	0.837	0.747	0.736	0.848	0.775	0.742	0.844	0.769	0.758
	Proposed	**0.848**	**0.822**	**0.815**	**0.886**	**0.817**	**0.827**	**0.880**	**0.784**	**0.842**	**0.872**	**0.795**	**0.806**	**0.879**	**0.793**	**0.825**
Dice score	**Segnet**	0.716	0.525	0.542	0.763	0.654	0.574	0.595	0.305	0.288	0.729	0.567	0.473	0.453	0.638	0.349
	Unet	0.904	0.870	0.840	0.903	0.860	0.824	0.886	0.836	0.795	0.895	0.850	0.797	0.891	0.85	0.818
	Proposed	**0.934**	**0.897**	**0.882**	**0.935**	**0.887**	**0.895**	**0.928**	**0.872**	**0.905**	**0.924**	**0.876**	**0.881**	**0.923**	**0.88**	**0.893**
MCC	**Segnet**	0.760	0.609	0.713	0.807	0.743	0.776	0.658	0.402	0.520	0.765	0.643	0.700	0.643	0.53	0/684
	Unet	0.935	0.861	0.850	0.933	0.871	0.883	0.922	0.861	0.879	0.930	0.876	0.892	0.923	0.879	0.894
	Proposed	**0.955**	**0.897**	**0.883**	**0.955**	**0.904**	**0.916**	**0.948**	**0.890**	**0.905**	**0.955**	**0.893**	**0.904**	**0.949**	**0.898**	**0.915**
Accuracy	**Segnet**	0.986	0.971	0.982	0.989	0.979	0.985	0.981	0.963	0.971	0.986	0.973	0.981	0.977	0.971	0.984
	Unet	0.990	0.992	0.991	0.996	0.991	0.994	0.991	0.990	0.993	0.991	0.992	0.990	0.991	0.991	0.990
	Proposed	**0.9961**	**0.986**	**0.962**	**0.996**	**0.991**	**0.996**	**0.996**	**0.991**	**0.994**	**0.995**	**0.996**	**0.991**	**0.995**	**0.995**	**0.993**

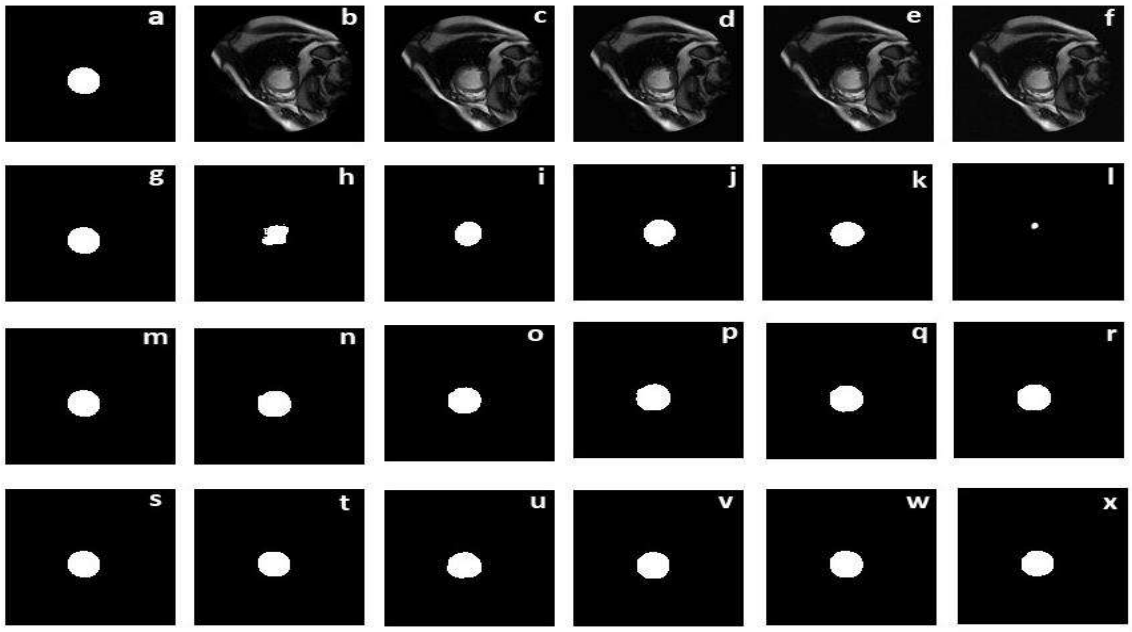

Figure 7.11: Segmentation results of SCD dataset. First column (a)-(s) presents the mask of the image being segmented, first row (b)-(f) presents the original image and noisy version (1%,3%,5% and 7%) of the original image. Second, third and fourth rows presents the segmented results for Seg-Net (h)-(l), U-net (n)-(r) and proposed network (t)-(x). Second, third, fourth, fifth and sixth column presents the segmented results for original (h)-(t), 1% (i)-(u), 3% (j)-(v), 5% (k)-(w) and 7%(l)-(x) noise corrupted images.

Table 7.6: Evaluation Metrics Comparison for SCD dataset

Evaluation Metric		**Images with % noise**				
		Original	**1%**	**3%**	**5%**	**7%**
Jaccard	**Segnet**	0.404762	0.39145	0.414578	0.399341	0.280637
	Unet	0.80289	0.792754	0.781819	0.776972	0.755668
	Proposed	**0.840893**	**0.828769**	**0.816192**	**0.81158**	**0.802293**
Dice score	**Segnet**	0.528635	0.521266	0.561425	0.527443	0.389564
	Unet	0.860453	0.85487	0.845826	0.83031	0.82461
	Proposed	**0.894782**	**0.880821**	**0.889817**	**0.875198**	**0.860907**
MCC	**Segnet**	0.619621	0.602494	0.590257	0.603583	0.546244
	Unet	0.873165	0.869793	0.860283	0.851614	0.832359
	Proposed	**0.910548**	**0.900022**	**0.89506**	**0.884658**	**0.867582**

Accuracy	**Segnet**	0.991418	0.990479	0.989066	0.991015	0.989112
	Unet	0.997342	0.998445	0.996254	0.997285	0.998141
	Proposed	**0.997789**	**0.998946**	**0.997865**	**0.998063**	**0.994**1

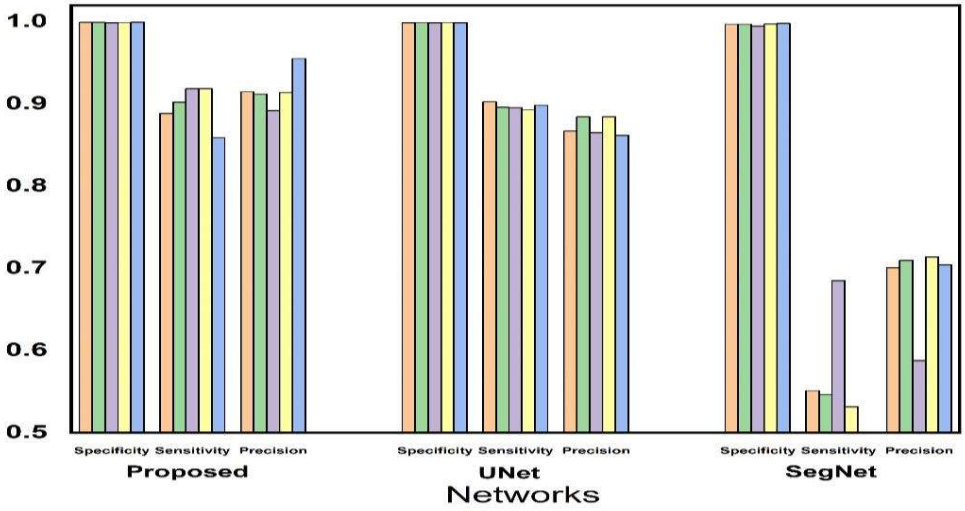

Figure 7.12: Comparison of Specificity, Sensitivity and Precision for the networks. First to fifth bars of each metric presents results for original image, 1%, 3%, 5% and 7% noise corrupted images respectively

Table 7.7: Comparison results of segmentation

Dice Score (ACDC Dataset)			
	LV	**MYO**	**RV**
Rohe et al[49]	0.929	0.868	0.881
Patravali et al[29]	0.920	0.890	0.865
Yang et al. [30]	0.919	0.806	0.818
Proposed	**0.934188**	**0.897406**	**0.882229**

Ablation Study:

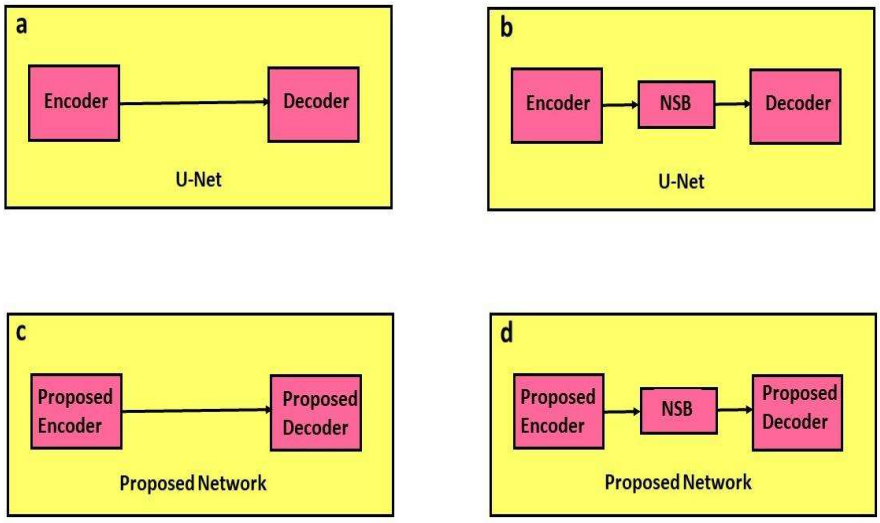

Figure 7.13: Layout plan of Ablation study

Table 7.8: Ablation Study: Evaluation Metrics Comparison of SCD dataset for noise free original images

Network Arrangement	Jaccard Index	Dice Score	MCC
(a)	0.80289	0.860453	0.873165
(b)	0.81983	0.86886	0.87946
(c)	0.82974	0.87602	0.88012
(d)	**0.840893**	**0.894782**	**0.910548**

The ablation study [51] was performed to study the network performance. Figure 7.8 shows the layout plan for the ablation study. This study was performed on SDC dataset for original, and 7% noise corrupted images. Figure 7.8(a) shows the conventional U-net as segmentation network, (b) shows U-net with NS block as segmentation network, (c) shows Modified proposed encoder-decoder network without NS block and shows Modified proposed encoder-decoder network with NS block. The changes observed for segmentation of images without noise and images with 7% added noise are shown in Table 7.8 and Table 7.9.

Table 7.9: Ablation Study: Evaluation Metric Comparison of SCD dataset for 7% noise corrupted images

Network Arrangement	Jaccard Index	Dice Score	MCC
(a)	0.755668	0.82461	0.832359
(b)	0.77082	0.83198	0.84054
(c)	0.78264	0.85035	0.84925
(d)	**0.802293**	**0.860907**	**0.867582**

The rows of Table 7.8 and Table 7.9 includes the changes in the network, while columns denote the comparison of evaluation metrics. Starting from U-net (Figure 7.8 (a)) to the final proposed method (Figure 7.8 (d)), the improvement in the evaluation metrics can be clearly observed. With every change in the U-net structure, the metrics tend to improve, which indicates the significance of every change made to the structure. The statistical paired t-test [52] was performed to validate the statistical significance of the proposed method. The null hypothesis was rejected as the value of p was less than 0.05 (with α =0.05) [53] [54], and a confidence level of 95% was achieved.

7.5 Discussion

7.5.1 Transfer learning Algorithm

Transfer learning uses the knowledge transfer phenomenon to segment the new image on which it is not trained. This eliminates the need of training the network from scratch and thus reduces the time and resources required. The transfer learning can be an efficient way to train a deep learning model for new tasks with faster convergence requiring lesser resources. In this view, we used available pre-trained model which is established to give accurate results on the ImageNet dataset. Feature pyramid network and U-Net [19] model architecture were used with encoders as DenseNet [20], ResNet [50], and VGG [51]. Data training pipeline includes region of interest (ROI) cropping, contrast limited adaptive histogram enhancement (CLAHE), normalization, and

random augmentation pre-processing of training images. During training, augmentation transforms were applied to each batch of data and are not stored to reduce training time. Cross-entropy is established to give good results in image segmentation but as it is limited to per pixel, the loss function used here is a weighted combination of cross-entropy and dice loss [43] [45].

Hyperparameters such as filter size, learning rate, optimizer selection to optimize the segmentation results are based on exhaustive methods of selection and then experiments are designed [58]. The initial learning rate varied from 10^{-3} to 10^{-4} and based on the learning curve, the learning rate was selected to be 10^{-4}. Optimizers such as Adam [59], SGD, and Adamax were tried for a total of 100 epochs and Adam was then selected based on dice score on the validation set. The training scheme described above helped in reaching the highest segmentation accuracy as dice-cross entropy loss decreased consistently on the validation set. Evaluation metrics values obtained for the network show that the algorithm worked properly giving good segmentation results.

Recently, U-Net and its modified version were mostly used for segmentation of CMR images and thus U-Net was used as a reference. Pre-trained models have updated network weights in place of randomly initiated weights if trained from scratch and thus lesser number of training iterations are required. Free-breath CMR images dataset available for training is limited and so we used the ACDC dataset with augmentation to include variability in training data along with an increment in the total number of images for training. A total of 6 transforms were applied to the training batch which consists of 8 images. Augmentation of training images leads to more generalization of models to predict CMR images alongside improved segmentation accuracy.

The accurate and precise segmentation of Cardiac MR images is highly desired for effective evaluation of pathological condition. Training of model from scratch requires high-end resources

and relatively more time as the model weights are randomly initialized. Using pre-trained model for transfer learning helps in reducing these requirements. The comparative analysis of segmentation results is shown in figure 7.8, 7.9, and 7.10 for End-Diastolic and End-Systolic cardiac phase. Performance parameters like Dice score, Jaccard Index are shown in subsequent figures and tables. According to results from ACDC challenge, the best results of mean dice score for LV, MYO, and RV class was 0.951, 0.917, and 0.933 respectively. The results obtained using proposed method is better and improved then state-of-the-art method available for cardiac MR image segmentation. The improvement in the dice score for the three classes are around 1% while a relatively huge decrease in directed Hausdorff distance is observed compared to state-of-art methods. Also, an overall reduction of about 30% training time is achieved for the method. The prominent figure of merit for the proposed network is the ability to produce exceedingly good results with training the network for as low as 50 epochs.

7.5.2 Modified U-Net Algorithm

The accurate and precise segmentation of medical images of various modalities is quite desirable for analyzing the regions affected by diseases. Real-time cardiac MR images are contaminated with spurious and unwanted Rician noise. These signals create issues for the segmentation networks in the segmentation of the boundary of the objects [55]. Thus the accurate analysis of the affected regions becomes difficult [56]. The proposed network eliminates the issue of noise in the segmentation process by the incorporation of noise stifler block [57]. This block is inserted between the encoder and decoder on the path, which transfers the feature and indices information. The noise stifler block (NSB) shown in figure 7.4 performs pixel-wise multiplication on the depth wise convolution outputs for the reduction of noise [58]. In NSB, depth wise separable convolution reduces the number of extra multiplication and produces strongly correlated features,

which are group normalized and then the pixel-wise multiplication is performed with the original feature maps. The features produced finally have reduced noise contents which are passed to the up-sampling decoder part. This process is missing in the conventional U-net, and our proposed network incorporates the change to produce more reliable results.

The comparative analysis of Figure 7.5 and figure 7.6 shows the proposed network's performance effectiveness. When the network was tested on a different dataset, the Jaccard and dice score [59] showed a good improvement of above 5% and 3.5%, respectively for noise-free images. For 7% noise, the improvement of above 6% and 4.8% was observed in both the metrics, respectively. This performance is achieved because of the incorporation of noise stifler block, which is missing in the conventional counterpart of the proposed network. The depth wise separable convolution produces enhanced spatial information as compared to the standard convolution because of reduced convolution cost. The use of group normalization [41] instead of batch normalization [60] gives stable accuracy as the dependence on batch size is eliminated. This combination produces robust feature maps with reliable information contents. These maps are further applied to NS blocks for suppressing the noise. In the delineation of boundary and contour details, the noise is the main disturbing element that is suppressed using the NS blocks. The noise suppressed features, when received by the up-sampling layers, produces the results with accurate boundary and contour details. The use of SELU activation instead of normal RELU gives the added advantage to saving the network from the problem of dying RELU [61], which in turn eliminates the issue of vanishing gradient. The self and internal normalization property of the SELU produces reliable normalized data, which produces more accurate results in comparison to the conventional counterparts [62]. The prominent figure of merit for the proposed network is the ability to produce exceedingly good results without retraining the network.

Table 7.1-7.2 depicts the improvement in the evaluation metrics. The increment in the Jaccard indicates that the amount of overlapping area is more between the segmented image and ground truth image for the proposed method as compared to the other methods in the study. The increased dice score signifies that the boundary pixels are segmented perfectly [63]. As the noise level is increased, the proposed method produces more clinically pertinent images as compared to the other methods in the study. This shows the better segmentation capability of the proposed network at various noise levels. The statistical analysis [64] shows a significant difference in the results obtained. The ablation study [65] shows the step by step increment of all the proposed changes in the conventional U-net model. The proposed changes show significant improvement one by one. Finally, the proposed method can be applied to the medical images, even corrupted by noise, for efficient segmentation.

7.6 Conclusion

The present work proposed a deep learning-based framework for the segmentation of cardiac MR images. The network not only accurately segmented the standard images but also effectively segmented the noisy images. Thus, it can be seen as the comprehensive capability of the network as the images acquired using any method contains noise that is unwanted. The region of interest in medical imaging is most important for planning the diagnosis of the disease. So, for medical analysis, the segments of the heart are clearly delineated by the proposed network, which is having nearly the same contour shape and boundary details. The network trained on one dataset produces accurate segmentation results for other datasets. In such a way, the need for retraining is eliminated by the proposed network. This proves the ambient capability of the network, which is helpful in saving time and resources. The proposed automatic approach can help in reducing the time and burden of the medical system by producing accurate results in less time.