A Light Weighted Deep Learning Framework for Multiple Sclerosis Lesion Segmentation

1st Palash Ghosal
dept.of Computer Science and Engg.
NIT Durgapur
Durgapur, India
ghosalpalash@gmail.com

2nd Pindi Krishna Chandra Prasad
dept.of Computer Science and Engg.
NIT Durgapur
Durgapur, India
pindikrishnachandra@gmail.com

3rd Debashis Nandi
dept.of Computer Science and Engg.
NIT Durgapur
Kolkata, India
debashisn2@gmail.com

***Abstract*—This paper presents a light weighted fully convolutional network based automated method for multiple sclerosis(MS) lesion segmentation from multimodal magnetic resonance (MR) scans which reduces the complexity of U-Net architecture and training time with significant extent. In multiple sclerosis problem, the discrepancy in texture, shape, form, size and location in MR images causes challenges in automatic subdivision of lesions. Other challenge we face, when we train these networks with unbalanced data. The MR images constitute unbalanced data because they contain more number of non-lesion class voxels than the lesion class voxels. If a network is trained with unbalanced data, the prediction with higher precision and lower recall can be made with a bias towards the non-lesion class. This is undesirable in the most medical diagnostic applications because false negatives are more serious than false positives. The proposed network has concentrated to solve these challenges to a desirable extent. The outputs of the network confirm that the recommended architecture is able to obtain higher accuracy and dice similarity co-efficient (DSC) with respect to the state-of-the-art.The performance of the networks is estimated by using randomized 5-fold subject-independent cross validation which helps us to judge the robustness and efficacy of the networks in MS lesion segmentation.**

***Index Terms*—multiple sclerosis, lesions, magnetic resonance imaging, deep learning, segmentation**

I. Introduction And Motivation

MS is a long-term, or persistent, condition that affects the central nervous system (CNS) which often leads to disability [1]. The CNS is made up of optic nerve, brain and spinal cord. Hardening of tissue in the body is defined as sclerosis. In MS, scar tissue appears in the CNS. This leads to travelling of messages between the brain and rest of the body in an uneven way. [2]. MR scanning of the brain has been a major advance in the diagnosis and can also help track the progression of MS. Fig.1 shows the images of MS in different modalities with the ground truth. MS has become a regular neurological disease taking in a set of people previously considered to be rarely affected all over the globe. Having affected 2-2.5 million people globally, the disease is being steadily identified in India largely due to inexpensive and easy availability of magnetic resonance imaging and increase in the quantity and quality of practicing neurologists [3]. The identification and treatment of MS occurs when the disease has already reached a relatively advanced stage in which motor symptoms are clear cut evident and substantial neurophysiological damage has occured already. At this point, any possibility of delaying disease progression or, attaining neuroprotection may already be remote. Till date, suboptimal referral of patients with MS to neurologists is a crucial barrier in the early diagnosis of MS in India. The experts widely agreed upon that timely referral and early initiation of treatment of MS are decisive for a reduction in the relapse rate and disability. Making an early diagnosis is of paramount importance so that progression of the disease and disability can be minimized. Detection of MS can be difficult and challenging, particularly in the early stage of the disease as the structure of the brain is complicated and can be analyzed only by experienced and highly skilled radiologists. In recent times a good amount of image-based research has utilized CAD systems in detection of neurodegenerative diseases such as Alzheimers disease and Parkinsonian syndromes [4]. MR slices are generally interpreted visually by medical practitioners. Most of the times it is performed manually by medical experts and often become tedious, time consuming and prone to human error. Therefore it would be beneficial to have a fully automated and real time computer aided diagnosis tool which might represent a useful tool for future applications in clinical practice for detecting MS. In the recent years, many review papers have briefed the technological advancements in medical image processing for MS lesion segmentation, leading to significant advances in this field, however with certain limitations. A variety of publicly available neuroimaging databases exist, including Brain Web published in 1997 [5], MICCAI 2008 [6] and MICCAI 2016 [7] that are being broadly used by many researchers for MS diagnosis. Supervised classification frameworks like decision random forests [8], k-nearest neighbours [9] have been applied extensively for MS lesion segmentation. In reference [10], a max-tree representation of MR slices were employed to highlight the high relative intensity region. In recent years, a lot of research is going on which apply machine learning and deep learning models which have been acquired for automatic extraction of features from the MRI to facilitate the treatment of MS patients by effective lesion segmentation and give promising results. A two-step cascaded Convolutional Neural Network (CNN) based architecture [16] has been proposed where two separate network has been trained with significant accuracy. The effect of intensity domain adaptation has also

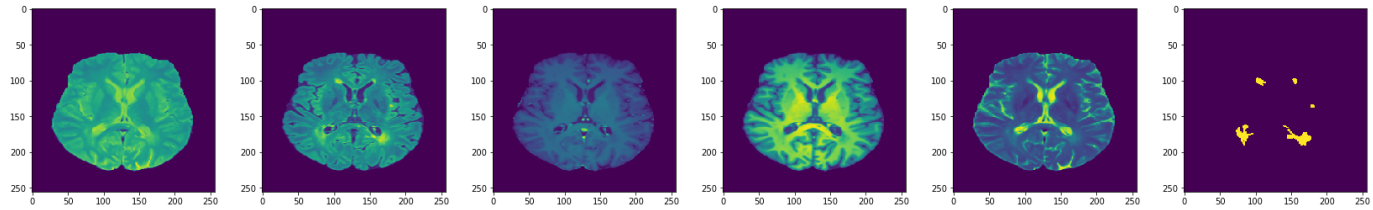

Fig. 1: Multi-modal MR Image for brain lesion segmentation.From left to right: MPRAGE, FLAIR, T1-w gadolinium enhanced, T1-w and T2-w/DP contrast enhanced and ground truth images respectively

been analysed on the proposed CNN architecture [11]. The endeavour of the work is to propose a light weighted CNN architecture without sacrificing two much accuracy. The main contributions of the present work is summarized as:

- The presented framework decreases the complexity of U-Net architecture by reducing the number of convolutional layers.
- We have reduced the number of filters as well as the number of concatenations of the convolutional layers in the contracting to the expanding path.
- We factually exihibit that the presented model overpasses some of the existing MS lesion segmentation techniques on various evaluation benchmarks.

The remaining part of this paper is structured as follows: Dataset description is given in Section II. Detailed explanation of each operation for our proposed methodology is given in Section III. The predicted output and their investigations is given in Section IV. Finally, the efficacy of the presented algorithm is concluded with some of the future scopes in Section V.

II. Dataset Description

A. Dataset Acquisition

We have trained the whole convolutional network end to end on MICCAI 2016 Dataset [7]. This dataset has MRI Scans of 15 patients acquired in different image domains with 5 different modalities T1-w, MPRAGE, FLAIR, T1-w gadolinium enhanced and T2-w/DP contrast enhanced images. All these modalities are of different resolutions for different patients. The size of each modality for first 5 patients is $(144 \times 512 \times 512)$, next 5 patients is $(128 \times 224 \times 256)$, rest 5 patients is $(261 \times 336 \times 336)$ respectively. The ground truth images of each training subject were provided by seven different human experts. The images were already provided with necessary pre-processing like image de-noising, intensity corrected and skull-stripped.

B. Preprocessing

1) Image Concatenation and Resizing: As the dataset consists of 3D images of 5 different modalities in Nifti format, We have concatenated the first slice of the one modality with first slice of other modalities, second slice of the one modality with second slice of other modalities till n^{th} slice of the one modality with n^{th} slice of other modalities. This concatenated slice contains the slice of all 5 different modalities. Therefore, the number of channels in the input image is equal to the number of modalities. This concatenated images of different patients are of different sizes. Therefore, We have re-sized all the images to $(256 \times 256 \times 5)$. The size of the input image is $(256 \times 256 \times 5)$.

2) Removing Null Data Samples: The prediction accuracy of a CNN architecture dependent on the training samples. Hence, repetitive and inapt data should be eliminated from training samples to reduce the quantity of features those exist in the training samples to decrease the time taken to construct the predictive model as well as enhance the predictive speed. So, we have removed all the null samples . If the sum of all voxel values of all the 5 channels of the input image is equal to zero , then that image and its corresponding ground truth is removed as it contains no information.

3) Scaling and Normalization: All these images are normalized to 0 to 1 improve training. Here normalization is done for all the slice of the different modality individually. Normalization of a particular image is done by subtracting the mean value of that image and dividing that by standard deviation of the slice of that particular modality.

1) Let the size of image of any modality of the MRI scan be $(h \times w \times c)$.
2) Scaling and normalization is operated on each channel independently.
3) Scaling on one channel in the range $[0, 1]$ is expressed as follows :

$$S(K) = \frac{K}{max(K) + \epsilon}$$

where K represents one channel of any modality of size $(h \times w)$

4) Normalization can be described as:

$$\text{Normalization}(K) = \frac{S(K) - \mu_{S(K)}}{\sigma_{S(k)} + \epsilon}$$

where μ and σ represent mean and standard deviation and $\epsilon = 0.00001$

III. Proposed Method

A. Network Architecture

In this method, we present a light-weighted U-net Architecture [12]. The CNN framework consists of 8 convolutional layers , 1 sigmoid layer at the end, 4 Max-Pooling layers, 4 Upsampling layers. The architecture consists of a encoder and

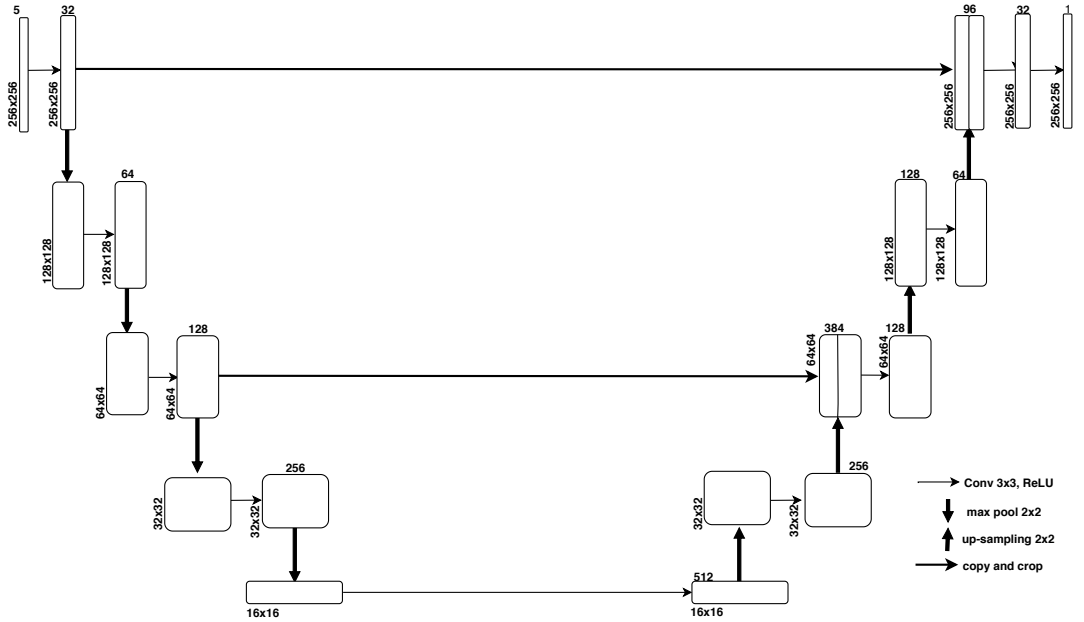

Fig. 2: Schematic diagram of the proposed light weighted U-Net architecture

a symmetric decoder. Each block in the encoder consists of a convolutional layer followed by batch normalization and by ReLU [13] activation function followed by a maxpooling layer to capture context . Each block in the expanding path is same as contracting path but it has upsampling layers that enables precise localisation instead of maxpooling layers. The central part between the contracting and expanding path contains 1 convolutional layer. Instead of concatenating all convolutonal layers in encoder to decoded like in U-net, we concatenated alternative convolutional layers. The output layer actually is a convolutional layer followed by a sigmoid activation function to detect whether there is lesion or not. We implement the CNN framework with the popular deep learning tool, Keras. Each block in the Network works as follows :

Initially, the input size = $(256 \times 256 \times 5)$ and convolutional kernel size in our network is (3×3) with strides (1×1) and same padding and maxpooling Kernel size is (2,2). We fed this input to the convolutional layer to extract the features.

The process of convolution [14] is described as follows: A 3D tensor with size $(256 \times 256 \times 5)$ is taken as the input in the first layer. The size of the convolution kernel is taken as a $3D$ tensor with size $(3 \times 3 \times 5)$. When we superimpose the kernel $(3 \times 3 \times 5)$ on top of the input $(256 \times 256 \times 5)$ at the spatial position $(0, 0, 0)$, the products of corresponding elements in all the 5 channels is computed and sum the $(3 \times 3 \times 5)$ products to get the convolution result at this spatial position. Then the kernel is moved by one element as the stride is $(1, 1)$ from left to right and from top to bottom to finish the convolution operation. In this operation, we used 32 kernels of size $(3 \times 3 \times 5)$. On convolution of one kernel with strides $(1, 1)$ and same padding on input tensor outputs a $2D$ tensor (256×256). This process is repeated for 32 kernels in the first block and the number of kernels in each convolutional layer increased exponentially in the contracting path. All these outputs of each convolution operation with each kernel is concatenated along the last dimension resulting in a $3D$ of size $(256 \times 256 \times 32)$. Therefore, in any layer, if the input tensor of shape $(h1 \times w1 \times d1)$ is fed into a convolutional layer of Kernel size $(h2 \times w2 \times d1)$, $d3$ number of filters, strides $(1, 1)$, same padding, Then it outputs a $3D$ tensor of shape $(h1 \times w1 \times d3)$. The mathematical equation of the convolution is defined as:

Let the input to the convolutional layer be a input of 3d shape $H \times W \times C$ and stride $(1, 1)$ with no padding, size of the kernel be $h \times w \times C$. Assuming D kernels are used, lets denote the kernels as f of size $h \times w \times C \times D$. Mathematically, the convolution operation in l^{th} layer can be expressed as,

$$y_{i^{l+1},j^{l+1},d} = \sum_{i=0}^{h} \sum_{j=0}^{w} \sum_{k=0}^{C} f_{i,j,k,d} \times x^{l}_{i^{l+1}+i,j^{l+1}+j,k} \qquad (1)$$

This equation is used $\forall\ 0 \leq d \leq D = D^{l+1}$, for any spatial position $\left(i^{l+1}, j^{l+1}\right)$ fulfilling $0 \leq i^{l+1} < H - h + 1 = H^{l+1}, 0 \leq j^{l+1} < W - w + 1 = W^{l+1}$ and $x^{l}_{i^{l+1}+i,j^{l+1}+j,d^{l}}$ denotes to the element of x^l indexed by the triplet $\left(i^{l+1} + i, j^{j+1} + j, k\right)$ and a bias b_k is usually added to $y_{i^{l+1},j^{l+1},k}$

Batch normalization is applied on the output of the convolutional layer. Each layer of a network learns by itself a little bit more when we use batch normalization. Its Algorithm is defined as follows:

Input: Let the value of x over a mini batch of size m be $\{X_1, X_2, ..., X_m\}$.
Output: $\{y_i = \text{Batch Normalization}_{\gamma,\beta}\,(X_i)\}$, where γ and β are trainable parameter.

$$\mu_B \leftarrow \frac{1}{m} \sum_{i=1}^{m} x_i \qquad (2)$$

$$\sigma_B^2 \leftarrow \frac{1}{m}\sum_{i=1} m\left(x_i - \mu_B\right)^2 \tag{3}$$

$$\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{4}$$

$$y_i \leftarrow \gamma\hat{x}_i + \beta \equiv \text{BatchNormalization}_{\gamma,\beta}\left(x_i\right) \tag{5}$$

The activation function Rectified Linear Unit (ReLU) is applied on the output of the batch normalization. It is used to truncate the every element in its input individually.

$$Y = max(0, x) \tag{6}$$

There are no parameters inside a ReLU layer. Maxpooling layer is applied on the output of ReLU layer. When we superimpose the kernel (2×2) on top of the output of ReLU layer at the spatial position $(0, 0)$, we find the maximum of the respective elements in all the channels to get the pooling result at this spatial postion. Then the kernel is moved by two elements as the stride is (2×2) from from left to right and top to bottom to finish the maxpooling operation. The number of channels of the input to the maxpooling layer will not be changed and but its dimensions (height and width) will be decreased. The mathematical equation of maxpooling layer is defined as follows:

1) Let the kernel size of the pooling taken in the CNN architecture be $(h \times w)$
2) Let the l^{th} layer be maxpooling layer and $x^l \in R^{H^l \times W^l \times C^l}$ be the input to maxpooling layer.
3) Here, stride is equal to pooling spatial extent.
4) The pooling layer output will be $3D$ data of size $y^l = \left(H^{l+1} \times W^{l+1} \times C^{l+1}\right)$
   where $H^{l+1} = \frac{H^l}{h}, W^{l+1} = \frac{W^l}{w}, C^{l+1} = C^l$

The pooling layer operates on its input $\left(x^l\right)$ channel by channel separately. In each channel $H^{l+1} \times W^{l+1}$ non overlapping sub-region are formed, each sub-region of size $h \times w$. maxpooling layer is represented as follows:

$$y_{i^{l+1},j^{l+1},k} = \max_{0 \le i < h, 0 \le j < w} x^l_{i^{l+1} \times h + i, j^{l+1} \times w + j, k} \tag{7}$$

where $0 \le i^{l+1} < H^{l+1}, 0 \le j^{l+1} < w^{l+1}$ and $0 \le k < C^{l+1} = C$

Upsampling layer is used in the expanding path of the network. This upsamples the convolutional layer output to a higher resolution. We have used the alternative concatenations of the output of the convolutional layer in contracting path to the corresponding input of convolutional layer in the expanding path.

B. Training Phase

The presented architecture is trained on the well known MICCAI 2016 data set of 15 Patients. After preprocessing the dataset as mentioned above, we have 2834 Samples of size $(256 \times 256 \times 5)$ which is divided into training and testing data. Initially the weights of all convolutional layers are initialized by xavier initialization. The presented method is trained on mini batch size of 32 with adam optimizer of learning rate 0.001. The loss function binary cross entropy is computed in this paper. We have trained the model for 60 epochs.

C. Testing and Validation Phase

To test the model, a random 5 fold cross validation strategy shuffles the entire dataset before training and validation so as to blindly randomize the dataset and then makes an $80:20$ split for each fold which may not occur in traditional 5 fold validation. The model is trained 5 times, using random 20% of all the data sub-samples as validation set for every fold. Rest of the dataset has the sole purpose of fitting the model, i.e., estimating the model parameters. The testing is done by evaluating the model with batch-size of 32. The network performance is evaluated by some of the very well known matrices like accuracy, dice coefficient, sensitivity, specificity. We demonstrated the average result of each metric and compared with some state of the art techniques in table II and also summarized the result in fig. 3

IV. Experimental Setup and Results

A. Implementation

The implementation of both the networks has been carried out with Keras using TensorFlow as backend. The network has been trained for 60 epochs using NVIDIA GTX 1070 8 GB GPU on a system with 16 GB RAM and a core i7 7th generation @ 4.0 GHz processor.

B. Comparison Metrics

To assess the automated methods presented for the segmentation of MS lesions a lot of measures have been used in the literature. These measures are based on comparing the result of the automated segmentation against the ground truth.Four widely used performance measures, namely Accuracy(Acc), DSC, Sensitivity(Sen) and Specificity(Spec) are used for evaluation purpose. They are expressed as follows:

$$DSC = \frac{2TP}{2TP + FP + FN} \tag{8}$$

$$sensitivity = \frac{TP}{TP + FN} \tag{9}$$

$$specificity = \frac{TN}{TN + FP} \tag{10}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where four basic retrieval terms FN = lesion regions wrongly classified as non-lesion regions, FP = non-lesion regions detected as lesion, TP = accurately classified lesion regions, TN = accurately classified non-lesion regions, .

C. Results

The proposed segmentation framework using binary cross entropy as loss function obtained the Accuracy of 96.79% , DSC of 0.76 , sensitivity of 0.65 and specificity of 0.86. We observed that there is no significant improvement in the training as we increase the number of convolutional layers for the MICCAI 2016 Multiple Sclerosis lesion segmentation dataset. Therefore, we limited the number of convolutional layer in each block to one. Additionally, we observed that there is no much improvement in the training after 40 epochs.

TABLE I: Layers disposal of the architecture of the proposed light weighted U-Net

Encoder				Decoder			
Layer	Fiter/K-Size/Stride	Trainable Parameter	input size	Layer	Fiter/K-Size/Stride	Trainable Parameter	input size
Conv0	32/ (3, 3) /1	1472	(256, 256, 5)	Conv9	1/ (1, 1) /1		(256, 256, 32)
BN0		128	(256, 256, 32)	Conv8	32/ (3, 3) /1	27680	(256, 256, 96)
Maxpool0	/ (2, 2) /2	0	(256, 256, 32)	UpSampling7	/ (2, 2) /	0	(128, 128, 64)
Conv1	64/ (3, 3) /1	18496	(128, 128, 32)	BN7		256	(128, 128, 64)
BN1		256	(128, 128, 64)	Conv7	64/ (3, 3) /1	73972	(128, 128, 128)
Maxpool1	/ (2, 2) /2	0	(128, 128, 64)	UpSampling6	/ (2, 2) /	0	(64, 64, 128)
Conv2	128/ (3, 3) /1	73856	(64, 64, 64)	BN6		512	(64, 64, 128)
BN2		512	(64, 64, 128)	Conv6	128/ (3, 3) /1	442496	(64, 64, 384)
Maxpool2	/ (2, 2) /2	0	(64, 64, 128)	UpSampling5	/ (2, 2) /	0	(32, 32, 256)
Conv3	256/ (3, 3) /1	295168	(32, 32, 128)	BN5		1024	(32, 32, 256)
BN3		1024	(32, 32, 256)	Conv5	256/ (3, 3) /1	1179904	(32, 32, 512)
Maxpool3	/ (2, 2) /2	0	(32, 32, 256)	UpSampling4	/ (2, 2) /	0	(16, 16, 512)
Conv4	512/ (3, 3) /1	1180160	(16, 16, 256)				
BN4		2048	(16, 16, 512)				

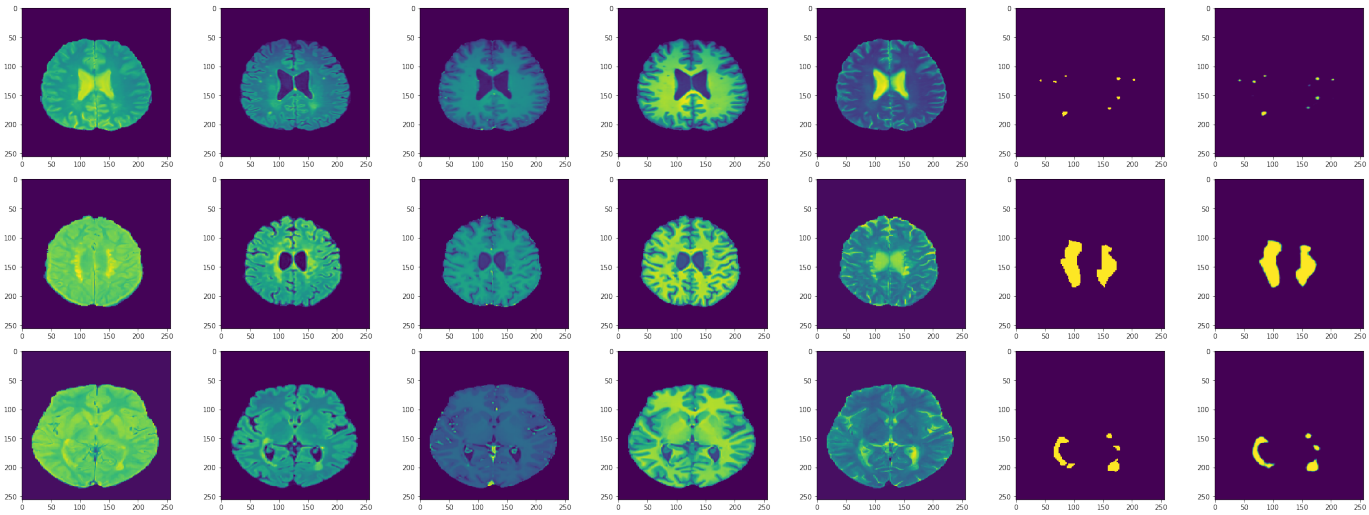

Fig. 3: From left to right: MPRAGE,FLAIR,T1-w gadolinium enhanced,T1-w and T2-w/DP contrast enhanced,ground truth and segmented output images respectively for three different patients.

TABLE II: Performance Analysis of the Proposed MS Lesion Segmentation Technique

Method	**DSC**	**Sensitivity**	**Specificity**	**Accuracy**
Andermatt et al [15]	0.63	0.54	0.84	92.07
Valverde et al. [16]	0.64	0.57	0.79	91.44
Birenbaum et al. [17]	0.63	0.55	0.80	91.26
Deshpande et al. [18]	0.60	0.55	0.73	89.81
Jain et al. [19]	0.55	0.47	0.73	88.74
Valcarcel et al. [20]	0.57	0.57	0.61	87.71
Sudre et al. [21]	0.52	0.46	0.66	86.44
Proposed	0.76	0.65	0.86	96.79

V. Discussion and Conclusion

Various CNN techniques have been recommended over the past few years for segmenting the Multiple Sclerosis lesions from the MRI Scans. The U-net, being once such architecture, has given remarkable results. But it is a heavy framework that takes quite some time and requires large GPU resources in training. However the framework which we proposed is more efficient in terms of resources as well as time taken. It is light weight modified U-Net which significantly reduced the complexity of the U-Net and gave a much better accuracy.The reason behind this is reduction in the number of convolutional layers in each block. Apart from this, We have also decreased the number concatenations of high resolution feedback from the encoder to decoder. We didnt have any fully connected layers and to maintain between the input and output dimensions, We have used the same padding. Over fitting is a problem that has been plaguing the field of biomedical image segmentation for quite long which we overcame by using batch normalization. This added noise to each hidden layers activation and had regularization effects. To sum up, we have presented a light weighted deep learning framework for the automatic MS lesion segmentation from the MRI which outperform the U-Net and few other architectures in terms of training time, accuracy and complexity. Despite the fitness of our approach compared to most of the state of the art methods, there exists several key points before the segmentation framework can be evolved into a fully functional clinical software.

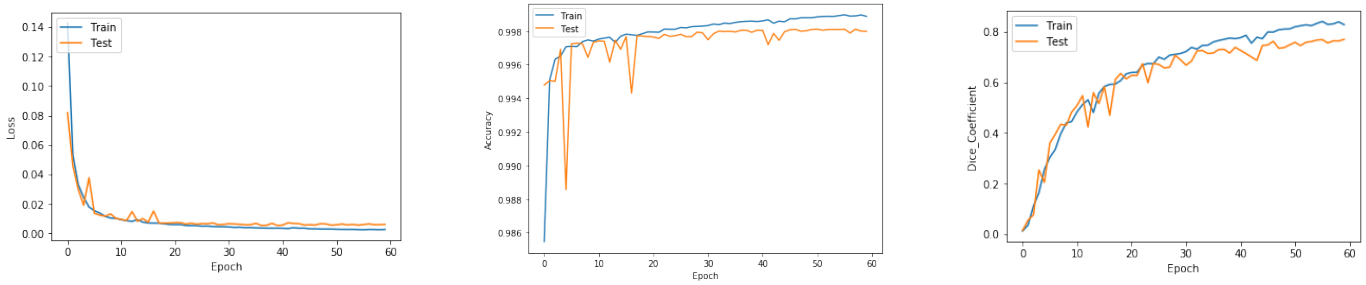

Fig. 4: From left to right: Plots of Loss Accuracy and Dice Coefficient for MS Lesion Segmentation on MSSEG 2016 dataset.

REFERENCES

[1] Dutta, R. and Trapp, B.D., 2011. Mechanisms of neuronal dysfunction and degeneration in multiple sclerosis. Progress in neurobiology, 93(1), pp.1-12.

[2] Greene, B.R., Healy, M., Rutledge, S., Caulfield, B. and Tubridy, N., 2014, August. Quantitative assessment of multiple sclerosis using inertial sensors and the TUG test. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 2977-2980). IEEE.

[3] Singhal, B.S. and Advani, H., 2015. Multiple sclerosis in India: An overview. Annals of Indian Academy of Neurology, 18(Suppl 1), p.S2.

[4] Ramrez, J., Grriz, J.M., Salas-Gonzalez, D., Romero, A., Lpez, M., lvarez, I. and Gmez-Ro, M., 2013. Computer-aided diagnosis of Alzheimers type dementia combining support vector machines and discriminant set of features. Information Sciences, 237, pp.59-72.

[5] Cocosco, C.A., Kollokian, V., Kwan, R.K.S., Pike, G.B. and Evans, A.C., 1997. Brainweb: Online interface to a 3D MRI simulated brain database. In NeuroImage.

[6] Styner, M., Lee, J., Chin, B., Chin, M., Commowick, O., Tran, H., Markovic-Plese, S., Jewells, V. and Warfield, S., 2008. 3D segmentation in the clinic: A grand challenge II: MS lesion segmentation. Midas Journal, 2008, pp.1-6.

[7] Commowick, O., Cervenansky, F. and Ameli, R., 2016. MSSEG challenge proceedings: Multiple sclerosis lesions segmentation challenge using a data management and processing infrastructure.

[8] Jesson, A. and Arbel, T., 2015. Hierarchical MRF and random forest segmentation of MS lesions and healthy tissues in brain MRI. Proceedings of the 2015 Longitudinal Multiple Sclerosis Lesion Segmentation Challenge, pp.1-2.

[9] Fartaria, M.J., Bonnier, G., Roche, A., Kober, T., Meuli, R., Rotzinger, D., Frackowiak, R., Schluep, M., Du Pasquier, R., Thiran, J.P. and Krueger, G., 2016. Automated detection of white matter and cortical lesions in early stages of multiple sclerosis. Journal of Magnetic Resonance Imaging, 43(6), pp.1445-1454.

[10] Salembier, P., Oliveras, A. and Garrido, L., 1998. Antiextensive connected operators for image and sequence processing. IEEE Transactions on Image Processing, 7(4), pp.555-570.

[11] Valverde, S., Salem, M., Cabezas, M., Pareto, D., Vilanova, J.C., Rami-Torrent, L., Rovira, ., Salvi, J., Oliver, A. and Llad, X., 2019. One-shot domain adaptation in multiple sclerosis lesion segmentation using convolutional neural networks. NeuroImage: Clinical, 21, p.101638.

[12] Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[13] Nair, V. and Hinton, G.E., 2010. Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning (ICML-10) (pp. 807-814).

[14] Wu, J., 2017. Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. Nanjing University. China, 5, p.23.

[15] Birenbaum, A. and Greenspan, H., 2016. Longitudinal multiple sclerosis lesion segmentation using multi-view convolutional neural networks. In Deep Learning and Data Labeling for Medical Applications (pp. 58-67). Springer, Cham.

[16] Valverde, S., Cabezas, M., Roura, E., Gonzlez-Vill, S., Pareto, D., Vilanova, J.C., Rami-Torrent, L., Rovira, ., Oliver, A. and Llad, X., 2017. Improving automated multiple sclerosis lesion segmentation with a cascaded 3D convolutional neural network approach. NeuroImage, 155, pp.159-168.

[17] Birenbaum, A. and Greenspan, H., 2017. Multi-view longitudinal CNN for multiple sclerosis lesion segmentation. Engineering Applications of Artificial Intelligence, 65, pp.111-118.

[18] Deshpande, H., Maurel, P. and Barillot, C., 2015. Classification of multiple sclerosis lesions using adaptive dictionary learning. Computerized Medical Imaging and Graphics, 46, pp.2-10.

[19] Jain, S., Sima, D.M., Ribbens, A., Cambron, M., Maertens, A., Van Hecke, W., De Mey, J., Barkhof, F., Steenwijk, M.D., Daams, M. and Maes, F., 2015. Automatic segmentation and volumetry of multiple sclerosis brain lesions from MR images. NeuroImage: Clinical, 8, pp.367-375.

[20] Valcarcel, A.M., Linn, K.A., Vandekar, S.N., Satterthwaite, T.D., Muschelli, J., Calabresi, P.A., Pham, D.L., Martin, M.L. and Shinohara, R.T., 2018. MIMoSA: an automated method for intermodal segmentation analysis of multiple sclerosis brain lesions. Journal of Neuroimaging, 28(4), pp.389-398.

[21] Sudre, C.H., Cardoso, M.J., Bouvy, W.H., Biessels, G.J., Barnes, J. and Ourselin, S., 2015. Bayesian model selection for pathological neuroimaging data applied to white matter lesion segmentation. IEEE transactions on medical imaging, 34(10), pp.2079-2102.