Lung Cancer Tumor Region Segmentation Using Recurrent 3D-DenseUNet

Uday Kamal, Abdul Muntakim Rafi, Rakibul Hoque and Md. Kamrul Hasan*

Abstract—**Lung cancer is one of the most severe and widespread that constitutes a major public health problem and has a high mortality rate. In this regard, proper segmentation of lung tumor from X-ray, Computed Tomography (CT scan) or, Magnetic Resonance Imaging (MRI) is the stepping stone towards achieving completely automated diagnosis system for lung cancer detection. With the advancement of technology and availability of data, the valuable time of a radiologist can be saved using computer tools for tumor segmentation. In this work, we present a data driven approach for lung tumor segmentation from CT scans by using Recurrent 3D-DenseUNet, a novel fusion of Convolutional and Recurrent neural network. Our approach is to train this network using image-volumes with tumor only slices of size $(256 \times 256 \times 8)$. A data-driven adaptive weighting method is also used in our approach to differentiate between tumorous and non-tumorous image-slices, which shows more promise than crude intensity thresholding of *0.70*, that we have used in this work for competition purpose. Our model has been trained and tested on the NSCLC-Radiomics dataset of 260 patients, provided by The Cancer Imaging Archive (TCIA) for 2018 IEEE VIP Cup. In this dataset, our proposed approach achieves an *average dice score* of 0.74, *mean surface distance* of 1.719 and *95% Hausdorff distance* of 7.249.**

Index Terms—**Lung Cancer, CT scan, Segmentation, 3D-UNet, Convolutional-LSTM, Recurrent 3D-DenseUNet, Dilation, Adaptive-weighting.**

I. Introduction

Lung cancer, also known as *lung carcinoma*, is a malignant lung tumor characterized by uncontrolled cell growth in tissues of the lung. The abnormal cells do not develop into healthy lung tissue, instead, they divide rapidly and form tumors. DNA mutation is the main cause of cancer and it can be caused by the normal aging process or through environmental factors, such as cigarette smoke, breathing in asbestos fibers, and to exposure to radon gas. Researchers have found that it takes a series of mutations to create a lung cancer cell [1]. The primary diagnosis of lung cancer is resulted from screening procedures, such as Computed Tomography (CT scan) or, Magnetic Resonance Imaging (MRI). The work of a radiologist is to confirm the existence of malicious cancer cells or tumors from the scanned images before conducting a more formal biopsy test. Radiomics is part of the job for a Radiologist. Radiomics refers to

All authors are with the Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology, Dhaka-1205, Bangladesh.

E-mail: *khasan@eee.buet.ac.bd

the process of extracting and analyzing several features (e.g., attenuation, shape, size, and location) from medical images with the ultimate goal of obtaining predictive or prognostic models. In this regard, Computer Aided Diagnosis (CAD) tools help the radiologist to accomplish this task of radiomics from the raw data of a CT scan.

A number of methods have been proposed in the literature to complete the task of medical image segmentation, thus helping the radiologist to indentify abnormal lesions or cancerous tumors. In [2], an image processing based technique has been applied to do lung tumor segmentation. Selin et al. in [2] has used morphological operations, filtering, seeding and thresholding as well as image residue to develop a system that automatically segments any lung tumor in a lung image, thereby achieving an accuracy of 97.14%. Besides lung tumor, a lung image contains many other components, or structures, such as clavicles and lungs. Therefore, detection of a lung tumor can be easily misguided by techniques that solely depend on image processing technique, where pixel intensity and thresholding are heavily utilized to specify the tumor region. Moreover, accuracy, sensitivity, specificity; these three metrics together define absolute measure of correctness of any segmentation related problem. In order to increase the robustness of tumor detection, aside from the pure image processing techniques, researchers are now more focused on the Machine Learning based techniques such as Deep Neural Networks(DNN) to solve medical image segmentation problems. In [3], automatic segmentation of brain lesions is achieved using Autoencoder. DNN is also used in tumor segmentation in [4]. In [5], lung cancer tumor detection and segmentation problem is tackled as a part of 2017 Kaggle Data Science Bowl Competition, where both signal processing techniques and neural networks are used to solve the problem.

Although CT scans are three-dimensional volumes and are usually anisotropic, however, in the literature, deep neural networks have been reported to perform well in 3D medical image segmentation [6]. In [6], Milletari et al. has introduced V-Net as well as an unique and well performed objective function to do volumetric segmentation from MRI. A successful adaptation of UNet from 2D to a 3D network has been also shown in [7], [8]. In [9], Liao et al. has demonstrated a 3D deep learning framework to perform automatic prostate segmentation. A more suitable network to work with is Hybrid-DenseUNet [10], where, complex connectivity of DenseNet [11] is adopted in the UNet architecture. As a result, these connections

convey associated input features into the deeper levels of the UNet architecture, which may be diminished due to forward propagation. However, a suitable cost function with appropiate constrain is equally important in training dedicated network. In [12], tversky loss function is introduced, which efficiently increases dice coefficient in solving segmentation problems. All in all, with a perfect combination of network and loss function, desired performance can be achieved.

In this work, we propose a novel Recurrent 3D-DenseUNet architecture in order to solve the problem asserted by the 2018 VIP-CUP challenge based on segmentation and prediction of Lung Cancer Tumor region via screening Computed Tomography or CT scans. Also, we use the tversky loss function described in [12] for training purpose. In this architecture, we have implemented Convolutional-LSTM Block within the encoder-to-decoder transition section in the UNet. As a result, this Conv2D_LSTM Block aggregates the inter-dependencies in the encoded outputs of the UNet, thus producing a better overall outcome in this problem.

The following sections of this paper explain the outline of our model, detailed architecture of the network, training procedure, and the results obtained after testing the model against the testing-data provided with the dataset.

Materials

A. *Description of Dataset*

In order to train our network, we have used the Dataset provided for the 2018 IEEE Video and Image Processing (VIP) Cup. The dataset is consisted of CT scans of 300 patients having a varying number of cross-sectional 2D images for each patient. These cross-sectional 2D image slices are generated from the DICOM files, which are provided as raw CT scan data. In addition, corresponding contour files for each patient are also provided in DICOM format along with the main dataset. Also, this dataset has been split into two non-overlapping groups of 260 patients and 40 patients, which have been specified to be used as training and validation data respectively. Aside from this dataset, a separate dataset of 40 patients is also provided by the 2018 IEEE VIP Cup organizers specifically for testing purpose.

B. *Data Generation*

As all the training, validation and testing data are provided in DICOM format, we have to convert each and every single DICOM file from *voxels* to world-coordinates and then normalize the data to generate 512×512– sized pixel images similar to the figures shown in Fig (1a) and (1b). We have also generated the corresponding binary masks of lung tumors (see Fig. (1c)) for training and validation slices from the DICOM files of contours. We have only extracted the tumor contour marked as *GTV1* from the various contour informations provided in the DICOM file format.
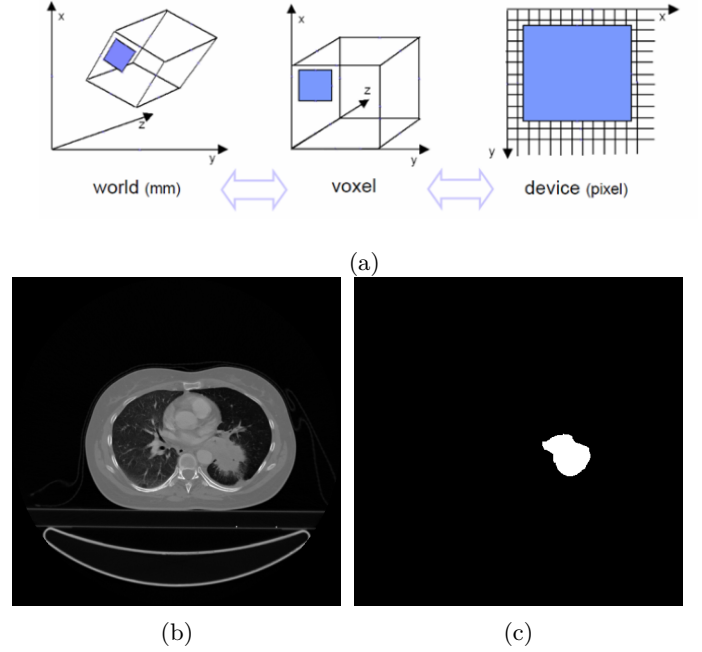

(a)

(b)   (c)

Fig. 1: Data generation steps from a DICOM file. **(a)** Conversion of DICOM files into pixel images. **(b)** A sample 2D-grayscale image of size 512×512 extracted from the DICOM files. **(c)** A binary mask of tumor for the image presented in *(b)*.

II. **Methods**

A. *Model Proposal*

The complete structure of our model is shown in Fig. 3, different parts of our model are outlined as follows:

1) First, we convert the raw DICOM files from world-coordinates to *voxels* to , then normalize the data and generate (512×512)– sized pixel images. We also extract their respective masks from corresponding RTconstruct files.

2) After extracting pixel images as well as their corresponding masks, we resize them to (256×256). However, only those images containing tumors, are selected to train our proposed architecture (shown in Fig. 3). In a group of *eight*, successive tumorous images are concatenated with a stride of *one*. As a result, we form the training dataset for our proposed Recurrent 3D-DenseUNet architecture having input size of $(256 \times 256 \times 8)$.

3) Next, we use a data driven approach to adaptively select an intensity threshold in order to distinguish between tumorous and non-tumorous slices. Here, from the intensity histogram of the 3D prediction output of the network, we have chosen a threshold for tumor detection. We have also tested on global threshold computed from the experimental results as well.

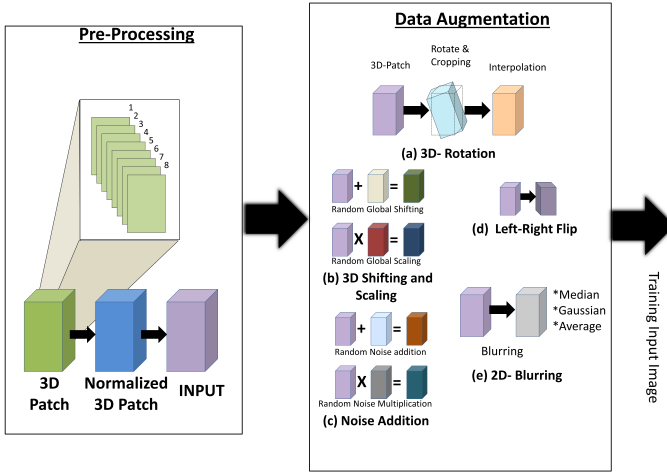4) During testing, all the CT scan slices of a single patient are taken at a time. At first, we generate

Fig. 2: Applied pre-processing and data augmentation methods.

TABLE I: **Achitecture of recurrent 3D-DenseUNet model**

ENCODER		DECODER	
Layers	**Specifications**	**Layers**	**Specifications**
Input	(None, 256, 256,8,0)	**Conv2D_LSTM Block**	(None, 64, 64, 8, 12, 0), # of filters = 256
Encoder Block (1)	(None, 256, 256,8,0), # of filters = 32	**Decoder Block (1)**	(None, 64, 64, 8, 12, 0), # of filters = 128
Max-Pooling Block (1)	(None, 128, 128.8,0), stride = 2	**Deconvolution Block (1)**	(None, 128, 128, 8, 0)
Spactial Dropout 3D (1)	(None, 128, 128.8,0), rate = 0.1	**Concatenation (1)**	(None, 128, 128, 8, 0)
Encoder Block (2)	(None, 128, 128.8,0), # of filters = 64	**Decoder Block (2)**	(None, 128, 128, 8, 0), # of filters = 64
Max-Pooling Block (2)	(None, 64, 64,8, 97,0), stride = 2	**Deconvolution Block (2)**	(None, 256, 256, 8, 0),
Spactial Dropout 3D (2)	(None, 64, 64,8, 22,0), rate = 0.1	**Concatenation (2)**	(None, 256, 256, 8, 0),
Encoder Block (3)	(None, 64, 64,8, 22,0), # of filters = 128	**Decoder Block (3)**	(None, 256, 256, 8, 0), # of filters = 32
Max-Pooling Block (3)	(None, 8,32, 32, 22, 0), stride = 2	**OutPut**	(None, 256, 256, 8, 0)
Spactial Dropout 3D (3)	(None, 8,32, 32, 22, 0), rate = 0.1		

overlapping $(256 \times 256 \times 8)$ volume of slices. We use our Recurrent 3D-DenseUNet network to generate a volume of binary masks. Then using the adaptive threshold selector, we remove some misdetections and generate a 2D binary mask for each slice. We also apply dilation on the mask slices using a (7×7) circular kernel to improve the IOU (Intersection over Union) value.

5) Finally we compute the required metrics:– dice-coefficient, hausdorff distance and 95% hausdorff distance for all the patients.

B. *Data Augmentation*

During the training of our proposed network, we have used several augmentation schemes to show different data to our neural network during each epoch. It helps to prevent over fitting. However, compared to the limited data resource available at this competition, data augmentation is an unavoidable choice to get better performance on unseen data. Thus we have used different kinds of augmentation at random order, which are presented in Fig. 2.

C. *Architecture*

1) Recurrent 3D-DenseUNet: In this work, we have proposed a novel architecture, that harnesses the core essence of 3 fundamental network, which are: DenseNet, UNet and Convolutional Recurrent Network. The basic architecture of our model has been presented in Table **I**.

UNet is a very popular and well-established network for medical image segmentation and classification problems. This competition is also related to medical image segmentation, and therefore it is an obvious choice for us to experiment with U-Net. Thus, in this architecture, we have adopted the UNet as the core skeleton of our network. Although 2D UNets are marked as compact network, nonetheless, degradation of features strength due to forward propagation of the input data is unavoidable

in any feedforward neural network unless skip connection technique is applied. As shown in [11], dense-blocks, which are the building blocks of DenseNet, are inter-connected with each-other in the network and thereby, prevent loss of acute statistical features from the dataset due to forward propagation. Inspired by this idea, we have applied a different type of encoder and decoder blocks in our network which is shown in Fig. 3. These encoder and decoder blocks are pseudo dense-blocks, which establish inter-connectivity between the input and middle layers of the block. Each encoder of our network consists of two subsequent 3D-Convolutional layers with $(3 \times 3 \times 3)$ kernel size, a Batch-Normalization layer, ReLU as an Activation layer and 2D-Maxpooling operation with kernel size of (2×2). The main reason for not using 3D-pooling operation is to preserve information along channel axis. A spatial dropout layer with dropout rate of 10% has been also used at the end of every encoder block. The main intuition of using volume data is to capture the inter-slice continuity of the tumor as a solid object. For this purpose, we have used Convolutional-LSTM network [13] within the transition section from encoder to decoder. With the increase in depth of convolutional layers, more and more higher level features are extracted from the input. To capture the inter dependencies at the highest level of our network, we have used the Conv_LSTM block at the transition section of encoder and decoder, which helps the network to identify inter-dependencies among the output features of the encoder, thus providing robust features for subsequent decoder blocks. Each decoder block consists of 2D-Deconvolutional through each channel, two subsequent 3D-convolutional layers, a Batch-Normalization Layer and ReLU as an Activation layer. At decoder block output, a skip connect is established with the corresponding previous encoder block.
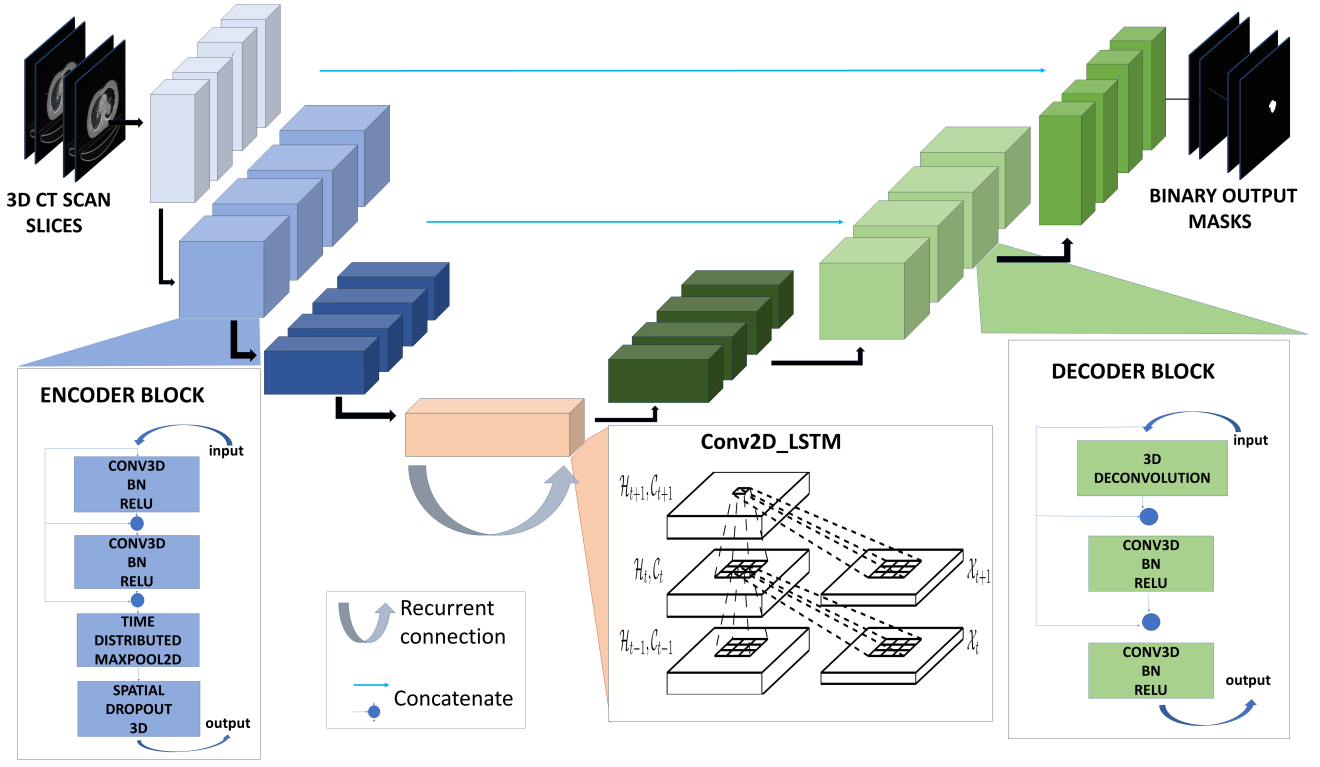
Fig. 3: Recurrent 3D-DenseUNet.

*2) **Adaptive Threshold Selector**:* Inspired by the idea of thresholding presented in [2], we have capitalized the idea of Ostu-thresholding [14] [15] to build a dynamic intensity threshold selector, which counts the histogram of the predicted 3D volume of tumor slices. Next, from the histogram, we calculate $3^{rd}, 4^{th}, 5^{th}$ and 6th highest peaks and corresponding intensities. These collected intensities are averaged using empirical parameters to predict a dynamic data driven threshold for each patient. A mathematical expression of the proposed heuristic process is given below:

$$I_d = \frac{1}{N-2}\Big[\sum_{i=3}^{N} a_i \times \arg I(h_i)\Big] \forall h_i \in \mathbf{H} \qquad (1)$$

where, N is the last index of the ordered histogram peaks, $\mathbf{H}$ is the histogram sorted in descending order, $argI(h_i$ indicates corresponding intensity at that specific h_i peak value, and lastly a_i are associated weighting coefficients for each peak value. We empirically set N = 6, and a_3, a_4, a_5, a_6 are set as 0.4, 0.3, 0.2, 0.1 respectively to calculate the data dependent intensity I_d. As shown in Fig. 4, we can clearly observe the variation of threshold in different patient.

Using this optimum intensity value I_d, we have been able to eliminate the unwanted seeds like tumor, that have been accidentally detected by our network, because, our network is trained only on the gold standard tumor only slices. As a result, the network detects a low intensified series of tumorlike seeds at different levels of the lung, which is basically a false detection by the network. This is

the exact reason why this adaptive threshold selector has been introduced to our model. As described before, this adaptive threshold varies patient to patient, thus, provides an opportunity to optimize the results.

III. **Computational Setup**

In this section, we discuss the training procedure in details. All of the experiments regarding training and implementation of the model are performed in hardware environments which included Intel Core-i7 8700K, 3.70 GHz CPUs and Nvidia GeForce GTX 1080 Ti (11 GB Memory) and Nvidia Titan XP (12 GB Memory) GPUs. The necessary codes were written in Python and, the neural network models were implemented by using the Keras API with Tensorflow in the backend.

All data provided in this competition have been in DICOM format, which are converted to 2D grayscale slices using Pydicom Library. We have also resized the slices from (512×512) to (256×256) to reduce memory consumption of the training device. We have used $(256 \times 256 \times 8)$ volume input instead of full lung volume to reduce training parameters as well as memory consumption.

We have trained our Recurrent 3D-DenseUNet model using an initial learning rate of 10^{-4} and batchsize of *2*. We have also used Learning Rate Scheduler and k-folding to improve the accuracy of the network. The dynamic threshold selector helps the network to subdue the effect of multiple seeds in the CT scan and increase the redundancy of the model. This heuristic approach fine-tunes the coefficients of the intensity histogram peaks
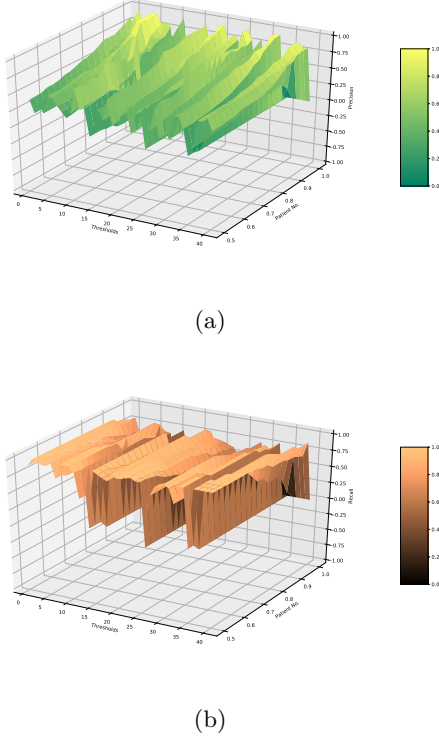
we average over the results of these 5 networks. This strategy proves to be much more efficient than training 5 different networks from scratch, because it needs much less time and is thus, much less intensive on resources. To converge to multiple local minima, we follow a cyclic annealing schedule as proposed by Loshchilov & Hutter in[16]. We lower the learning rate at a very fast pace, encouraging the model to converge towards its first local minimum after as few as 10 epochs. The optimization is then continued at a larger learning rate, which perturbs the model and dislodges it from the minimum. We repeat this process several times to obtain multiple convergences. The learning-rate schedule that we use is a type of Cyclic Exponential Annealing. The learning-rate $\alpha(t)$ for the t-th epoch is defined as:

$$\alpha(t) = \alpha_0 . k^{mod(t-1,M)} \qquad 0 \leq t \leq n.M \qquad (3)$$

where α_0 is the initial learning rate at the start of each cycle, M is the number of epochs in each cycle, n is the number of cycles the model is trained and k is the annealing factor such that $k \in (0,1)$. In other words, we split the training process into n cycles, each of which starts with a large learning rate, which is annealed to a smaller learning rate. The large learning rate $\alpha(1) = \alpha_0$ provides the model enough energy to escape from a critical point, while the small learning rate $\approx \alpha_0 . k^M$ drives the model to a well behaved local minima. For our experiments, we have used $\alpha_0 = 10^{-4}, k = 0.5, M = 30$ and $n = 5$.

IV. **Results and Discussions**

This is the core section of our paper. Here, we have presented a complete summary of the associated experimental results as well as brief reasoning to establish our arguments. But beforehand explaining the details, in the following subsection; we have presented the basic evaluation criteria, under which- our model has been tested.

A. *Evaluation Criteria*

In this work, the generated segmented contours are compared against the manual contours for all test images using evaluation metrics similar to the following ones.

1) Dice Coefficient: This is a measure of relative overlap, where 1 represents perfect agreement and 0 represents no overlap.

Evaluation Criteria 1:
$$D = \frac{2(|X| \cap |Y|)}{|X| + |Y|} \qquad (4)$$

where $\cap$ denotes the intersection operator, X and Y are the ground truth and test regions. Please note that dice coefficient (D) has a restricted range of $[0,1]$. As recommended in literature of image validation, a good overlap occurs when $D > 0.700$. The following two conventions are considered in computation of Dice coefficient: (i) For True-Negative (i.e., there is no tumor and the processing algorithm correctly detected the absence of the tumor), the dice coefficient would be 1, and; (ii) For False-Positive (i.e.,

(a)

(b)

Fig. 4: **(a)** Recall variation in different thresholds for different patients. **(b)** Precision variation in different thresholds for different patients.

calculated from the overall detected tumor slices by the network. In the following paragraphs, we have discussed some fine tuning techniques used to improve the output of our segmentation model.

Loss Function

In our experiment, we have used the Dice Coefficient equation to construct our loss function as $(1 - dice\_coef)$. The equation is given below:

$$loss = 1 - \frac{2(|A| \cap |B|)}{|A| + |B|} \qquad (2)$$

During training, we have minimized the loss function and have saved the weight when the *dice_coeff* was maximized. Thus the best saved weight is used for result generation.

Cyclic Learning-Rate Scheduler

In this method, we train our Recurrent 3D-DenseUNet model using a cyclic learning-rate schedule. As the network converges to a local optima, we take a snapshot of the weights of the network. Then, the learning rate is changed abruptly at the end of a cycle. As a result, the network jumps out of the current minima and converges to a new one as training progresses. In this way, we take 5 snapshots of the network while training, which correspond to 5 local minima of the error manifold. Finally, during testing,

there is no tumor but the processing algorithm mistakenly segmented the tumor), the dice coefficient would be 0.

*2) **Mean surface distance**:* The directed average Hausdorff measure is the average distance of a point in X to its closest point in Y, i.e.,

$$\vec{d}_{H,avg}(X,Y) = \frac{1}{|X|} \sum_{x \in X} \min_{y \in Y} d(x,y) \qquad (5)$$

The (undirected or) average Hausdorff measure is the average of the two directed average Hausdorff measures given by

Evaluation Criteria 2:

$$d_{H,avg}(X,Y) = \frac{\vec{d}_{H,avg}(X,Y) + \vec{d}_{H,avg}(Y,X)}{2} \qquad (6)$$

In short, this parameter gives a pseudo measure of specificity of the tumor detection. Generally, specificity refers to the exactness of detection in segmentation problems.

*3) **Hausdorff distance (95% Hausdorff distance)**:* The directed percent Hausdorff measure, for a percentile r, is the rth percentile distance over all distances from points in X to their closest point in Y. For example, the directed 95% Hausdorff distance is the point in X with distance to its closest point in Y is greater or equal to exactly 95% of the other points in X. In mathematical terms, denoting the rth percentile as K_r, this is given by

$$\vec{d}_{H,r}(X,Y) = K_r \left(\min_{y \in Y} d(x,y) \right) \forall x \in X \qquad (7)$$

The (undirected) percent Hausdorff measure is defined again with the mean:

Evaluation Criteria 3:

$$d_{H,r}(X,Y) = \frac{\vec{d}_{H,r}(X,Y) + \vec{d}_{H,r}(Y,X)}{2} \qquad (8)$$

Similarly, from the intuitive sense, it is pretty much clear that this *95% Hausdorff distance* a similar parameter to sensitivity, where it physically calculated the maximum separation of detected and ground truth tumor region. Therefore, reducing this parameter will increase the performance of the system

In order to produce a benchmark result in this dataset, we have generated the binary masks for each patient from the lateral slices of CT scan images– using our Recurrent 3D-DenseUNet architecture and best trained weight. However, we have only passed exactly 60% slices from the middle of the CT scan and passed it through our network. This gives us a volume of detected tumor slices, which has been passed to the dynamic intensity selector to eliminate the extra tumor like seeds from the CT scan segment. Volume causes continuity of the tumor and thus causes extra detection of tumor. Thus we use this adaptive weighting method which specifically identifies the tumor containing slices. 3D input volumes helps the network to detect slide to slide tumor connectivity from CT scan, however it also increases miss-detection rate.

As a false detection in one slice may continue to exist in subsequent slices due to this connective detection method, we have decided to not only depend on the network, but also on the morphological adaptive thresholding to reduce false detection rate.

There are three specific parameters which are presented neat and cleanly in the paper. Besides, necessary codes and files are attached with this report for further evaluation. We have evaluated the performance of our algorithm on the provided validation dataset and the result is shown in Fig. 5.

We have applied different methods to produce the required results, and depending on the variation of these results, we have decided our final model. A full analysis of different methods applied in our network is presented in Table II.

From Table II, we can clearly see the trade off among the results, predicted by different thresholds. Here, we can see, that both the adaptive weighting method with morphological dilation and crude threshold of 0.70 without dilation produces similar results on dice-coefficient. However, the adaptive method increase the amount of false-negative compared to 0.70 threshold but reduces false-positive number of slides. As lung tumor segmentation is considered as a most preliminary step of lung cancer screening. So, instead of emphasizing on extra detection, we have given more importance to prevent miss detection because it better to stay safe than sorry. Thus, we have adopted a global threshold of 0.70 in our pipeline. But, using our dynamic thresholding algorithm, if one can find the optimal threshold, it is quite possible to balance the trade of prefectly between false-positive and false-negative. Also, we can clearly see the best result in 0.70 threshold with dialation system from Table II, where we have achieved an overall dice score 0.740% , mean surface distance of 1.791 and 95% Hausdorff distance of 7.249 on the validation data. We have applied dilation on the tumorous slices, because, due to crude thresholding, there is a strong possibility that some part of the tumor may have been diminished. Morphological operation such as dilation expands the tumors along the borderline and thus, improves the dice-coefficient.

V. **Conclusion**

In this paper, we have proposed a hybrid 3D Recurrent encoder-decoder based Convolutional Neural Network architecture for accurately detecting and segmenting lung tumor from CT scan. Here we have used the recurrent layer at the deepest part of our network (after the end of encoding and before decoding) where it learns the inter-slice dependency at the reduced feature domain. We have also added short intra block skip connections between the input and mid-layers of every encoder and decoder block to recover spatial information, which may be lost during convolution operation. We have trained this model using 3D volume data with various types of data augmentation to prevent over-fitting. This pipeline shows an overall dice score 0.740% , mean surface distance of 1.791 and 95%
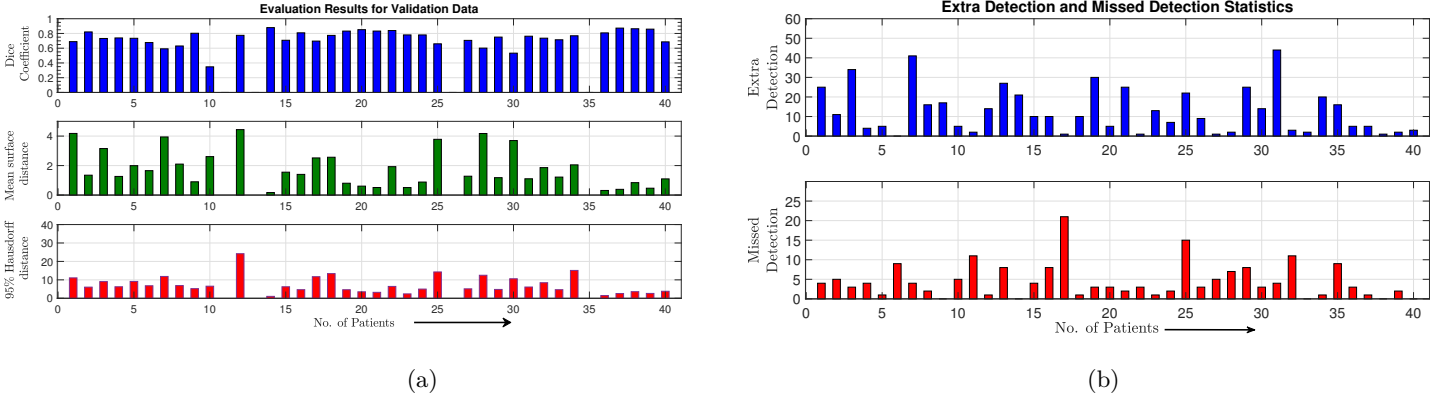
Fig. 5: **(a)** Results of the three evaluation criteria for the validation data are presented in this barchart. **(b)** No. of slice for extra-detection and missed-detection for each patient are presented in the above barchart.

TABLE II: **Comparison between different methods applied**

Patient ID	Criteria 1 (dice score)	Criteria 2 (mean surface distance)	Criteria 3 (95% Hausdorff distance)	False Detection (in no. of slides)	Failed Detection (in no. of slides)
0.80 Threshold	0.675	2.019	7.150	278	297
0.70 Threshold	0.706	1.910	5.320	316	262
0.50 Threshold	0.700	2.280	8.680	823	156
No Threshold	0.675	5.967	19.38	823	139
Adaptive Threshold with dialation	0.706	1.564	6.022	245	466
0.70 Threshold with dialation	0.740	1.791	7.249	316	260

Hausdorff distance of 7.249 on the validation data provided for the IEEE video and image processing cup 2018. Our training pipeline involves cyclic learning rate scheduler, which helps the model to converge to the global minima. During testing, we have used overlapping 3D patches from the CT scan volume and generate the final prediction mask by averaging similar slices. Also, the use of data-driven adaptive intensity selection approach has helped us to get the initial view of our final pipeline, which helped us to select a global intensity threshold value of 0.70 for this model. This has promoted us to achieve a better segmentation dice score as well as helped us to reduce the overall number of false positive and false negative no. of slices in the validation data. In fine, more modifications and improvements can be done in this model, especially in the neural network section, which may lead to successful implementation of our Recurrent 3D-DenseUNet in other medical imaging sectors as well.

Acknowledgment

REFERENCES

[1] Apar Pataer, Neda Kalhor, Arlene M Correa, Maria Gabriela Raso, Jeremy J Erasmus, Edward S Kim, Carmen Behrens, J Jack Lee, Jack A Roth, David J Stewart, et al. Histopathologic response criteria predict survival of patients with resected lung cancer after neoadjuvant chemotherapy. *Journal of Thoracic Oncology*, 7(5):825–832, 2012.

[2] Selin Uzelaltinbulat and Buse Ugur. Lung tumor segmentation algorithm. *Procedia Computer Science*, 120:140–147, 2017.

[3] Tom Brosch, Lisa YW Tang, Youngjin Yoo, David KB Li, Anthony Traboulsee, and Roger Tam. Deep 3d convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation. *IEEE transactions on medical imaging*, 35(5):1229–1239, 2016.

[4] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31, 2017.

[5] Kingsley Kuan, Mathieu Ravaut, Gaurav Manek, Huiling Chen, Jie Lin, Babar Nazir, Cen Chen, Tse Chiang Howe, Zeng Zeng, and Vijay Chandrasekhar. Deep learning for lung cancer detection: Tackling the kaggle data science bowl 2017 challenge. *arXiv preprint arXiv:1705.09435*, 2017.

[6] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 565–571. IEEE, 2016.

[7] 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.

[8] Hao Chen, Qi Dou, Lequan Yu, and Pheng-Ann Heng. Voxresnet: Deep voxelwise residual networks for volumetric brain segmentation. *arXiv preprint arXiv:1608.05895*, 2016.

[9] Shu Liao, Yaozong Gao, Aytekin Oto, and Dinggang Shen. Representation learning: a unified deep learning framework for automatic prostate mr segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 254–261. Springer, 2013.

[10] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng Ann Heng. H-denseunet: Hybrid densely connected unet for liver and liver tumor segmentation from ct volumes. *arXiv preprint arXiv:1709.07330*, 2017.

[11] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, volume 1, page 3, 2017.

[12] Seyed Raein Hashemi, Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, Sanjay P Prabhu, Simon K Warfield, and Ali Gholipour. Tversky as a loss function for highly unbalanced image segmentation using 3d fully convolutional deep networks. *arXiv preprint arXiv:1803.11078*, 2018.

[13] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[14] Liu Jianzhuang, Li Wenqing, and Tian Yupeng. Automatic thresholding of gray-level pictures using two-dimension otsu method. In *Circuits and Systems, 1991. Conference Proceedings, China., 1991 International Conference on*, pages 325–327. IEEE, 1991.

[15] Miss Hetal J Vala and Astha Baxi. A review on otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 2(2):pp–387, 2013.

[16] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.