

ENSEMBLE APPROACH FOR IMPROVING KIDNEY TUMORS SEGMENTATION PERFORMANCE ON CT IMAGES USING DEEP LEARNING MODELS

Geethanjali T M

Assistant Professor, Department of Information Science, P.E.S. College of Engineering,
Mandya, Karnataka, 571401, India
geethanjalitm@pesce.ac.in

Minavathi

Professor, Department of Computer Science, P.E.S. College of Engineering,
Mandya, Karnataka, 571401, India
minavathi@pesce.ac.in

Dinesh M S

Honorary Professor, P.ET Research Center,
Mandya, Karnataka, 571401, India
dineshmys@gmail.com

Abstract

Clinical picture examinations are conducted with the help of image segmentation, which allows the computerized picture split into a set of pixels. In segmentation, the aim is to enhance and modify the delineation of a picture so that it becomes more distinguished and easier to investigate. Kidney growths address a sort of malignancy that individuals of old age are bound to create. In this respect, Deep Learning (DL) models are becoming increasingly appealing. Creating models for kidney tumor segmentation assist doctors/radiologists in recognizing cancers with effective division as an integral step. A comparison of the segmentation approaches using Attention U-Net, Feature Pyramid Network (FPN) and LinkNet Models is presented in this paper to develop the ideal prediction model on Computed Tomography (CT) images. Various encoders are used in all three architectures to build different predictor models. Ensemble approach using Attention-U-Net architecture outperforms compared to FPN and LinkNet architectures with IoU scores 95.66 % (kidney) and 93.86% (tumor).

Keywords: Deep Learning; Kidney Tumor Segmentation; Attention U-Net; Feature Pyramid Network; LinkNet; Ensemble; Computed Tomography.

1. Introduction

The most frequently diagnosed genitourinary cancer is renal cell carcinoma [1]. Identifying kidneys and tumors accurately from medical images, such as CT scan is crucial for appropriate treatment. The cancer of the kidney develops from the kidney cells and may spread slowly or precipitously. It usually appears as single mass, but different types of tumors can occur in any kidney. It enables doctors to make more accurate treatment plans by examining the segmentation of kidney and tumors. Yet, segmenting the region of interest (ROI) manually can be tedious and time consuming task because radiologist need to tag out ROI of all the slices for each individual. Therefore nowadays precise auto segmentation tools are thus deeply needed.

Kidney cancer is ranked 9th in men and 14th in women [2, 3]. Affected individuals are much more likely to be cured if they are diagnosed and treated as soon as possible. Radiologists evaluate images of multiple anatomical structures to detect fine changes that may indicate disease and aid in the diagnosis of certain illness. At the same time, there has been a tremendous growth in the amount of data derived from CT, which is why there is growing need by many researchers to develop a reliable automated system to assist radiologists in diagnosing and planning

treatment. Manual measurements, however, are tedious and can lead to inter-operator and intra-operator variability.

A great deal of research effort has been devoted to the development of novel deep learning methods for automatically segmenting kidneys and kidney tumors from CT images. Majority of early works are centered on the use of unsupervised training methods. It is difficult to segment kidneys and its tumors since the location, size and shape vary considerably across patients. In this regard, the following key considerations are: First, where the tumor arises from can vary greatly with each individual. The tumor may grow inside or outside the organs. Furthermore, both the shape and size of tumors are highly variables. Tumors may represent very small areas within the kidneys in some patients and vastly larger areas over the entire kidney in others. They can also exhibit a regular shape, a distorted one or a very scattered one. Moreover, tumor tissues are heterogeneous, which means that CT images can attribute diverse intensity values to them. Lastly, segmentation of kidneys and tumors is not straight forward which in turn introduces additional obstacles since there is large background consisting of multiple labels.

The main objective of the work is to segment kidney and its tumors pixel wise effectively. In the work, focus is on using ensemble weighted averaging method to improvise the evaluation score and to get better prediction compared to individual models.

2. Related Works

Deep learning methods have emerged recently as promising tools for analyzing medical images. One among these methods is Convolutional Neural networks (CNN) [14, 15, 16, 17, 18, 19, 20] architectures which have already proven superior to traditional algorithms in a variety of computer vision exercises. The Fully Convolutional Networks (FCN) [21] offer a powerful system for labeling images at the pixel level and U-Net [22] models are also widely used. In spite of the availability of various deep learning techniques used for image segmentation, medical images diverge drastically from natural images. As a result, medical images are different from natural images in several significant ways: initially the homogeneity of medical images reduces the likelihood of detecting disease progression across individual images. Furthermore, there is no sharp border between organs, tumors or other areas of interest.

3D U-Net architecture and auto encoder architecture proposed by Prashant Jadiya [5] aimed in segmenting kidneys and tumors simultaneously. The three-dimensional U-Net model was not able to predict tumors isolated however, the auto-encoder architecture was better able to predict tumors in isolation. A method developed by M.Haghighi et al. [6] uses Convolutional neural networks fed from two, three dimensional inputs simultaneously to learn kidney segmentation tasks both spatially and time-wise. Segmentation performance was tested on normal and diseased kidneys in pediatric patients with varying degrees of hydronephrosis. The mean dice coefficient was 91.4 in the normal kidneys and 83.6 in the abnormal kidneys. Luana Batista da Cruz et al. [3] describe an automated method for determining the CT slice segmentation of the kidney using two CNN models and post-processing step. The First model, AlexNet was used for CT slice classification, while the second model used for determining the segmentation of the kidney.

In an effort to identify malignancies in the kidney, Mredhula.L et al. [7] developed an algorithm using fuzzy C-means clustering to perform semi-automated segmentation, which showed promising results. Using a SC-UNet connected cascade, Chenxia Wang et al. [8] have developed a method to segment kidney and tumors using shape and context information in 3D U-Net architecture. The first SC-UNet handles the rough kidney segmentation, the second SC-UNet handles the smooth kidney and tumor segmentation. Seda Arslan Tuncer et al. [9] have proposed a novel kidney segmentation technique that uses the spine as a reference point, Using the Connected Component Labeling (CCL) and K-means on the dataset, the k-means method outperformed the CCL approach.

As discussed, a variety of deep learning techniques have been employed to segment kidneys and tumors. We are currently exploring how to ensemble pre-trained models to analyze abdominal images and segment kidneys and tumors automatically.

3. Data and Experimental Setup

3.1. Dataset

The dataset consists of 210 CT volumes in Neuroimaging Informatic Technology Initiative (NIFTI) format obtained from the KiTS19 Challenge [4]. Ground truth images clearly define kidney tumor voxels in all of these CT scans in the late arterial phase where white corresponds to kidney, gray to tumor and black has backgrounds which are 512x512 and in grayscale. KiTS19 dataset statistics used for experimentation uses 35401, 4425 and 4426 slices for training validating and testing the models respectively.

3.2. Data Preparation and Pre-Processing

From the abdominal CT volumes included in KiTS19 challenge, 44,252 images were used for training, validation and testing the models. Among the records, 10% of the scans are used as the test dataset, while the rest are divided into training (80%) and validation (10%) datasets respectively. To standardize the input images, the HU

values range of [-1500, 2000] were considered and the data was formatted to 128x128x3. All the images were converted to PNG format. The following algorithm was used for data preparation

Algorithm: Data Preparation Pseudo code

1. Loading the Volumetric image (3D).
 2. Splitting the Volumetric images to 2D slices.
 3. Normalize the images.
 - I. Set each pixel values of the image to -1500 (≤ -1500) and 2000 (others).
 - II. Subtract -1500 from each image value and divide it by range(max_value – min_value)
 4. Saving each of the image and mask slices in PNG format.
-

As a result of encoding, the ground truth image has 0, 1, and 2 values for background, tumor and kidney respectively.

3.3. Experimental Setup

This section outlines the experimental setup used for kidney and its tumor segmentation. Based on the memory constraints of the graphics card, a unique configuration of encoders and architectures are used to build a model taking into account the batch size of 8 with learning rate 1e-4. Models are evaluated taking imagenet pretrained weights. Using Ubuntu 20.04 as the operating system, the experiment was performed on a workstation containing Nvidia Quadro RTX 5000 16GB graphics card.

4. Methods

This section is dedicated to interpret the problem of segmentation, description of the architectures with encoders and ensemble model used to improve the IoU score.

4.1. Formalization

For experimentation, let dataset D consists of images $X \in \mathbb{R}^{nr, nc}$ and masks $Y \in \{0, 1, 2\}^{nr, nc}$ where $nr=128$, $nc=128$. Suppose $M_{archi, encod}$ represents a deep learning model for segmentation, in which *archi* denotes architecture and *encod* denotes encoders. All architectures are experimented using ImageNet weights. In the segmentation models encoders are used for the extraction of features. The dataset used in the training phase are D_{train} , X_{train} and Y_{train} which are subsets of D , X and Y . The validation ($D_{validation}$) and test (D_{test}) subsets of dataset D comprising of X_{val} , Y_{val} and X_{test} , Y_{test} respectively. The objective of the present study was to find a point of optimal performance of the model $M_{archi, encod}$ during training considering the input X_{val} such that prediction (\hat{Y}_{val}) and the target (Y_{val}) outputs are as closely as possible. Implementation is based on the model that performs the minimum validation error out of all epochs is selected for testing on unseen dataset D_{test} ($D_{validation} \cap D_{test} \cap D = \emptyset$). Fig.1 presents a model based on training of encoder with an architecture used for kidney tumor segmentation. X is the input passed to the architecture, calculating \hat{Y} (predicted output) where the arrows represents the forward and backward stages of the model. A number of times, training samples are used to improve Loss (\mathcal{L}) in response to the loss calculation using \hat{Y} and Y (target output). The Loss is then propagated to weights and this process gets repeated until \mathcal{L} get converge.

4.2. Architectures with encoders

This section describes the model to be tested using three architectures Attention U-Net [10], Feature Pyramid Network [11] and LinkNet [12]. Using soft attention gates, Attention U-Net [10] model is trained by using different weights to various parts of an image based on relevance to learn how to focus on target structures of different sizes and shapes. The model was mainly applied to medical imaging. FPN architecture [11] applies a 1x1 convolutional layer and adds the features instead of copying and appending them as in U-Net architecture. A LinkNet [12] utilizes residuals in each encoder and decoder block and is partially transferred from the general network. Table 1 shows encoders used with their variations denoted in paranthesis corresponding to Attention U-Net, LinkNet and FPN Architectures.

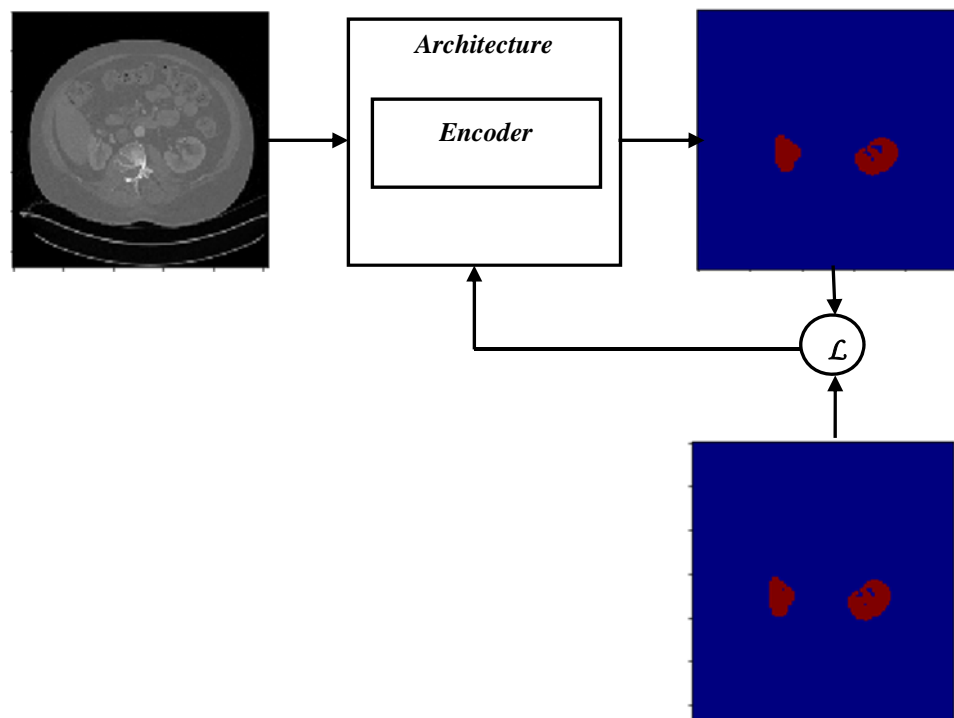


Fig. 1. Model for kidney tumor segmentation with an encoder-based architecture

Table 1. Encoders used in Experimentation for feature extraction

Attention U-Net	FPN	Link Net
VGG (16,19)	VGG (16,19)	VGG (16,19)
Dense Net (121, 169, 201)	Dense Net (121,169,201)	Dense Net (121,169,201)
Efficient Net (b0-b7)	Efficient Net (b0-b7)	Efficient Net (b0-b7)
ResNet (50,101,152)	Inception (resnetv2,v3)	Inception (resnetv2,v3)
ResNetv2 (50,101,152)	Mobile Net (v2)	Mobile Net (v2)
	ResNet (18,34,50,101,152)	ResNet (18,34,50,101,152)
	SeresNet (18,34,50,101,152)	SeresNet (18,34,50,101,152)
	SeresNext (50)	SeresNext (50)

In evaluating the networks by increasing the depth and examining small convolutional filters, VGG [13] was used to optimize classification on large scale images named subsequent to the Visual Geometry Group and leading to better performance. While ResNets [14] facilitates a residual learning structure for training enhancement and simplifies the optimization process to yield greater performance in image recognition. DenseNet's [15] feed forward connection enables the establishment of connectivity from every layer to the next by reusing and strengthening features for boosting the performance. In addition, MobileNet [16] has been found to be an efficient object detection light weighted model taking into account depth seperable convolutions. Adding up EfficientNet Family [17] have concern with the effects of balancing width, depth and resolution factors, whereas the Seresnet [18] version focuses on channel interactions by using the squeeze and excitation block. Finally, inception version resnetv2 [19] and v3 [20] uses the estimation of residual and non residual blocks for object recognition.

4.3. Ensemble model

A number of predictors are combined in the ensemble approach to outperform the best individual predictors. The whole approach is satisfactory when the predictors are unbiased to one another. Practice is made using different algorithms to get different results. This will increase the likelihood of getting more accurate results. When few

accurate predictors have been constructed and have been combined to give a better predictor, the ensemble strategy is used. Since there is no universal model developed to handle images with different properties, ensemble approach is used in our work. The Pseudo code used in ensembling approach is described below

Algorithm Pseudo code used in Ensemble Approach

1. Load different models used in ensemble process only for prediction.
2. Create the list of models.
3. Convert the list of predicted models to arrays.
4. Grid search for the best combination of weights that gives maximum IoU Score.
5. Finally, all models used in this approach multiplies the prediction value with the best weights and adds together to get maximum pixel value.

Top six models with the utmost Mean IoU value on a test dataset in each architecture were ensembled together to progress the outcomes as shown in Table 3. In this study, weights are added to the model to raise IOU score and better segment kidneys and tumors.

4.4. Evaluation Metric

In order to assess the performance of the model, comparison is made between the predicted segmentation mask (P) and its corresponding target segmentation mask (Y) using Intersection over the Union (IoU) and Dice-Coefficient metrics which is given in Eq. (1) and Eq. (2) :

$$IntersectionOverUnion(IoU) = \frac{P \cap Y}{P \cup Y} \quad (1)$$

$$Dice\ Coefficient = \frac{2 * (P \cap Y)}{P + Y} \quad (2)$$

The Loss function for FPN and LinkNet Architectures is determined by a combination of dice loss and focal loss with equal weights, and the Categorical Cross Entropy for Attention U-Net architectures.

5. Results and Discussions

Illustration how actually the IoU scores can be improved by ensembling the top six models using the three architectures with multiple encoders individually is discussed in this section. Table 2 presents the results of 75 models build using three architectures. The results depicted in Table 3 clearly indicate the reason for proposing the ensemble approach for segmenting kidney and Tumors. Ensemble Attention U-Net model gives better IoU scores with precise segmentation of kidneys with their tumors. As a result of this study, the Deep learning model designers will be able to compare their novel models against these three models, in order to confirm and evaluate the performance. The experiment on initializing encoder weights confirms that pretrained weights of imagenet improves segmentation results significantly in Attention U-Net compared with FPN and LinkNet Architectures.

Table 2. Architecture-Encoder Results

Architecture Encoders		Validation Set				Testing Set			
		Mean IoU	Background IoU	Tumor IoU	Kidney IoU	Mean IoU	Background IoU	Tumor IoU	Kidney IoU
Attention U-Net	vgg16	0.95531	0.99954	0.9270	0.9393	0.95584	0.99955	0.9288	0.9390
	vgg19	0.94903	0.99941	0.9237	0.9239	0.94728	0.99941	0.9199	0.9224
	resnet50	0.95660	0.99963	0.9214	0.9486	0.95794	0.99964	0.9247	0.9494
	resnet101	0.95377	0.99962	0.9150	0.9466	0.95533	0.99963	0.9192	0.9470
	resnet152	0.95450	0.99962	0.9169	0.9469	0.95385	0.99962	0.9144	0.9474
	resnet50v2	0.94301	0.99954	0.8946	0.9348	0.94283	0.99954	0.8950	0.9339
	resnet101v2	0.93848	0.99950	0.8849	0.9310	0.93940	0.99950	0.8884	0.9302
	resnet152v2	0.94561	0.99955	0.9007	0.9365	0.94480	0.99955	0.8998	0.9350
	densenet121	0.96021	0.99965	0.9295	0.9514	0.96064	0.99966	0.9309	0.9512
	densenet169	0.95883	0.99963	0.9273	0.9495	0.95811	0.99965	0.9251	0.9495
	densenet201	0.95697	0.99962	0.9239	0.9473	0.95583	0.99962	0.9210	0.9467
	efficientnetb0	0.95928	0.99964	0.9271	0.9510	0.95967	0.99965	0.9281	0.9511
	efficientnetb1	0.95914	0.99964	0.9282	0.9495	0.95834	0.99964	0.9250	0.9503
	efficientnetb2	0.96183	0.99965	0.9336	0.9521	0.9607	0.99965	0.9307	0.9519
	efficientnetb3	0.95953	0.99965	0.9284	0.9505	0.9603	0.99965	0.9299	0.9514
	efficientnetb4	0.96175	0.99966	0.9328	0.9527	0.96057	0.99960	0.9297	0.9523
	efficientnetb5	0.95728	0.99963	0.9234	0.9487	0.95857	0.99964	0.9269	0.9491

	efficientnetb6	0.96220	0.99966	0.9338	0.9530	0.96167	0.99966	0.9326	0.9527
	efficientnetb7	0.95817	0.99964	0.9253	0.9495	0.95822	0.99965	0.9245	0.9504
FPN	vgg16	0.33049	0.99149	0.0	0.0	0.33054	0.99164	0.0	0.0
	vgg19	0.33049	0.99149	0.0	0.0	0.33054	0.99164	0.0	0.0
	resnet18	0.63724	0.99801	0.0	0.9137	0.63695	0.99794	0.0	0.9129
	resnet34	0.82424	0.99694	0.7898	0.6859	0.82692	0.99686	0.8133	0.6706
	resnet50	0.88201	0.99913	0.7614	0.8854	0.88008	0.99910	0.7581	0.8829
	resnet101	0.89926	0.99916	0.8122	0.8863	0.90039	0.99917	0.8152	0.8867
	resnet152	0.91754	0.99932	0.8464	0.9068	0.91449	0.99932	0.8394	0.9047
	seresnet18	0.92515	0.99938	0.8604	0.9156	0.92737	0.99940	0.8677	0.9150
	seresnet34	0.89430	0.99900	0.8092	0.8745	0.89516	0.99906	0.8116	0.8748
	seresnet50	0.91858	0.99935	0.8450	0.9113	0.92254	0.99937	0.8570	0.9111
	seresnet101	0.90124	0.99916	0.8154	0.8891	0.89647	0.99913	0.8041	0.8861
	seresnet152	0.90592	0.99922	0.8249	0.8935	0.90487	0.99923	0.8225	0.8928
	seresnext50	0.90829	0.99928	0.8203	0.9052	0.91250	0.99930	0.8332	0.9049
	mobilenet	0.92381	0.99938	0.8582	0.9137	0.92856	0.99940	0.8712	0.9150
	mobilenetv2	0.92421	0.99937	0.8589	0.9142	0.92707	0.99939	0.8673	0.9144
	inceptionv3	0.92537	0.99941	0.8587	0.9179	0.92896	0.99942	0.8694	0.9180
	Inception resnetv2	0.93642	0.99945	0.8858	0.9240	0.93952	0.99946	0.8942	0.9248
	densenet121	0.63591	0.99803	0.0	0.9097	0.63530	0.99796	0.0	0.9079
	densenet169	0.60142	0.99709	0.0	0.8071	0.59678	0.99691	0.0	0.7934
	densenet201	0.64092	0.99810	0.0	0.9246	0.64037	0.99802	0.0	0.9230
	efficientnetb0	0.92201	0.99938	0.8512	0.9154	0.92287	0.99939	0.8557	0.9134
	efficientnetb1	0.92532	0.99932	0.8666	0.9100	0.92820	0.99934	0.8774	0.9078
	efficientnetb2	0.93448	0.9994	0.8819	0.9220	0.93292	0.99945	0.8770	0.9222
	efficientnetb3	0.93375	0.99940	0.8838	0.9180	0.93634	0.99943	0.8894	0.9201
	efficientnetb4	0.93859	0.99945	0.8912	0.9250	0.93718	0.99946	0.8878	0.9242
	efficientnetb5	0.92531	0.99938	0.8625	0.9140	0.92427	0.99939	0.8601	0.9132
	efficientnetb6	0.75571	0.99328	0.7670	0.5067	0.76014	0.99371	0.7714	0.5152
	efficientnetb7	0.93251	0.99944	0.8749	0.9232	0.93409	0.99945	0.8793	0.9234
LinkNet	vgg16	0.93356	0.99942	0.8806	0.9205	0.93230	0.99943	0.8783	0.9191
	vgg19	0.92279	0.99936	0.8552	0.9137	0.92178	0.99937	0.8528	0.9131
	resnet18	0.91122	0.99917	0.8427	0.8917	0.91102	0.99918	0.8420	0.8918
	resnet34	0.90356	0.99915	0.8226	0.8888	0.90358	0.99916	0.8239	0.8876
	resnet50	0.91736	0.99933	0.8415	0.9112	0.91513	0.99933	0.8366	0.9093
	resnet101	0.89589	0.99912	0.8038	0.8846	0.89221	0.99912	0.7926	0.8849
	resnet152	0.90274	0.99922	0.8117	0.8972	0.90283	0.99923	0.8114	0.8978
	seresnet18	0.90789	0.99920	0.8307	0.8937	0.90816	0.99919	0.8337	0.8915
	seresnet34	0.88424	0.99904	0.7648	0.8888	0.89153	0.99909	0.7852	0.8903
	seresnet50	0.91877	0.99931	0.8497	0.9072	0.92021	0.99934	0.8524	0.9088
	seresnet101	0.91166	0.99927	0.8328	0.9028	0.91263	0.99929	0.8362	0.9023
	seresnet152	0.92290	0.99930	0.8576	0.9117	0.92445	0.99936	0.8622	0.9117
	seresnext50	0.9290	0.99933	0.8499	0.9134	0.92307	0.99935	0.8552	0.9146
	mobilenet	0.92345	0.99933	0.8602	0.9107	0.92482	0.99936	0.8652	0.9098
	mobilenetv2	0.91539	0.99928	0.8431	0.9037	0.91747	0.99930	0.8483	0.9048
	inceptionv3	0.87942	0.99907	0.7489	0.8902	0.88164	0.99907	0.7556	0.8902
	Inception resnetv2	0.92794	0.99936	0.8708	0.9136	0.93052	0.99939	0.8781	0.9140
	densenet121	0.91715	0.99925	0.8496	0.9025	0.91671	0.99926	0.8490	0.9018
	densenet169	0.89049	0.99908	0.7888	0.8835	0.88853	0.99908	0.7828	0.8836
	densenet201	0.90499	0.99921	0.8203	0.8954	0.90374	0.99923	0.8191	0.8928
	efficientnetb0	0.91651	0.99929	0.8448	0.9053	0.91610	0.99931	0.8429	0.9060
	efficientnetb1	0.92367	0.99935	0.8594	0.9122	0.92473	0.99937	0.8623	0.9125
	efficientnetb2	0.92430	0.99934	0.8621	0.9114	0.92685	0.99935	0.8707	0.9104
	efficientnetb3	0.92740	0.99937	0.8673	0.9155	0.92885	0.99939	0.8711	0.9159
	efficientnetb4	0.92923	0.99939	0.8692	0.9190	0.92983	0.99941	0.8710	0.9190
	efficientnetb5	0.92871	0.99940	0.8677	0.9189	0.92919	0.99941	0.8694	0.9186
	efficientnetb6	0.92836	0.99941	0.8681	0.9175	0.92640	0.99942	0.8632	0.9165
	efficientnetb7	0.92984	0.99942	0.8697	0.9203	0.93052	0.99943	0.8721	0.9199

Table 3. Architecture Ensembled Results

Architecture	Ensemble Models	Kidney IoU Score	Tumor IoU Score
Attention U-Net	densenet121 efficientnetb0 efficientnetb2 efficientnetb3 efficientnetb4 efficientnetb6	0.9566	0.9386
FPN	Inception-resnetv2 efficientnetb2 efficientnetb3 efficientnetb4 efficientnetb7 inceptionv3	0.9310	0.9057
LinkNet	vgg16 inception-resnetv2 efficientnetb3 efficientnetb4 efficientnetb5 efficientnetb7	0.9286	0.8994

Table 4 outlines how attention U-Net with different encoders actually converges faster when compared to FPN and LinkNet architectures. This table gives summary of Convergence time Approximation (in seconds) of few encoders which are used in all the three architectures. According to the findings of all three architectures, Attention U-Net architecture is the most promising one regarding transfer learning, which in turn improves performance (Table 3) and boosts convergence (Table 4).

Table 4. Convergence time approximation(in seconds) of three Architectures

Encoder	Converge Time (Attention U-Net)	Converge Time (FPN)	Converge Time (LinkNet)
Dense-net121	8280	12240	17160
Dense-net169	8381	9660	13240
Resnet 50	6840	7200	12096
Efficient net-b0	7770	13560	12400
Efficientnet-b1	7888	14340	14448

Ensembled outcomes for few slices of all the three architectures are presented in Fig.2, which indicates difference in segmentation quality between the best (Attention U-Net) and worst (LinkNet) architectures. All models appear to generate similar results in terms of Mean IoU, Kidney and Tumor IoU values, but qualitative analysis shows that the ensemble model has boosted the Mean, Kidney and Tumor IoU scores. Two columns in Fig.2 indicate the input image and target mask, while the remaining three columns indicate the results of architectures. The mean, Background, Tumor and kidney IoU scores obtained are presented below each image slice. In comparison with other architectures, it is evident from the first and last image rows that the results achieved using ensemble Attention U-Net model are better. The second row image clearly shows that the segmentation outcomes are obtained only in ensemble Attention U-Net architecture. Whereas, ensemble FPN and LinkNet architectures fails to segment tumors which is presented in third row.

The Fig. 3 illustrates how ensemble model boost the performance when compared to the top six individual models used in Attention U-Net architecture assembly. Taking into account the complete testing dataset,fewsllices considered in Fig.3 shows ensemble model efficiency is superior. Values written in column 2-8 denotes tumor and kidney IoU scores respectively. As a result, the ensemble model with other models used for assembling provides better IoU Scores of tumors and kidneys. Same applies to FPN and LinkNet architecture. Therefore, ensembling plays an important role in segmentation.

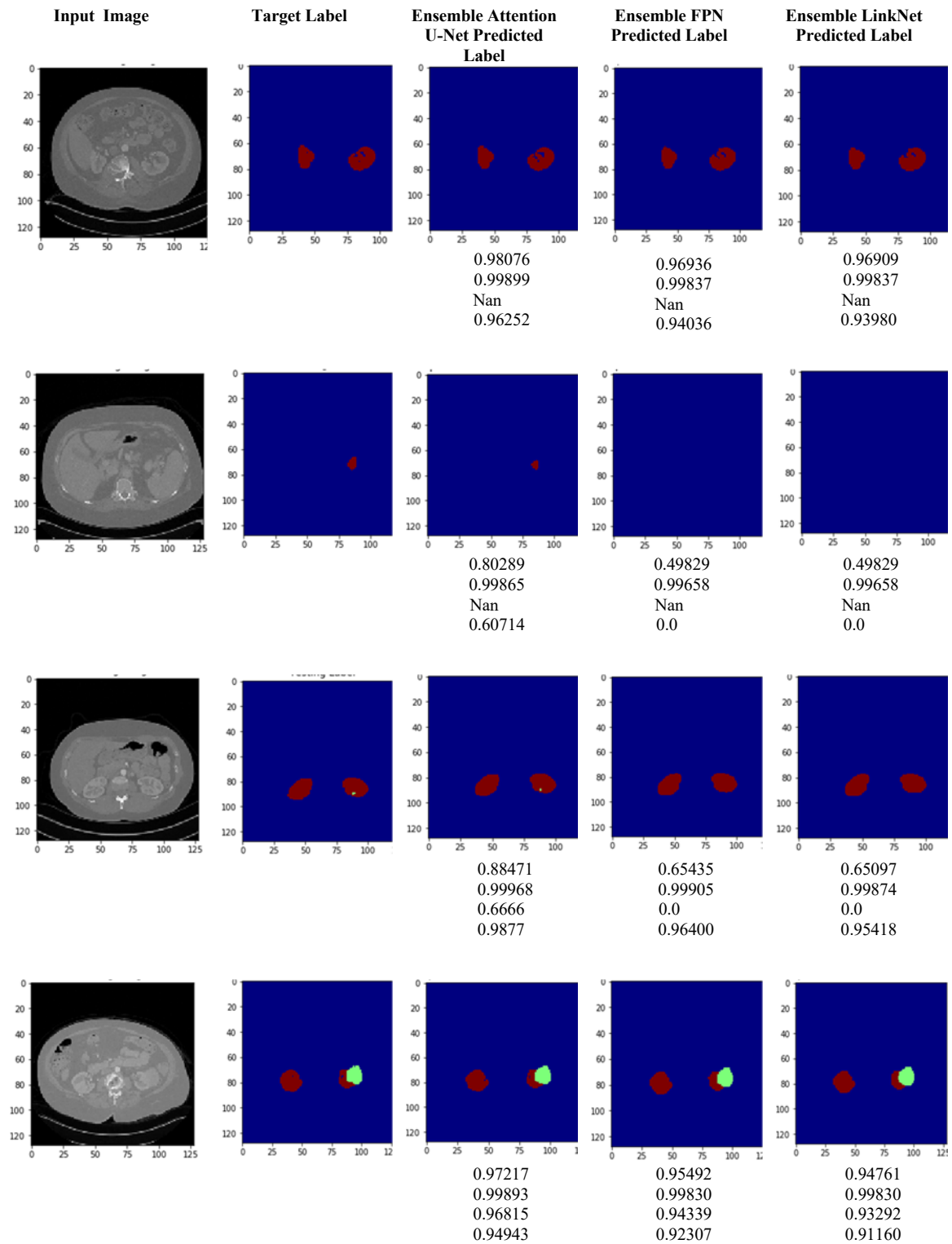


Fig.2. Ensemble outcomes from three architectures

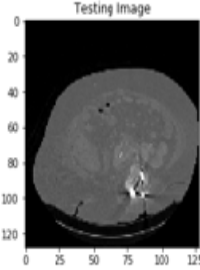
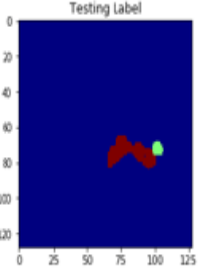
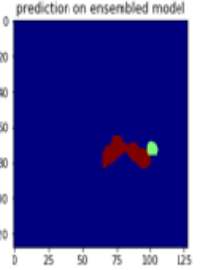
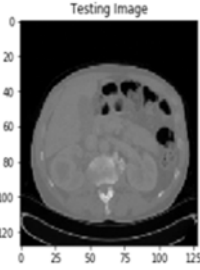
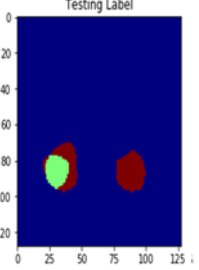
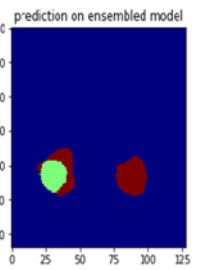
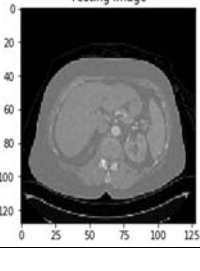
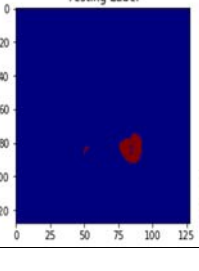
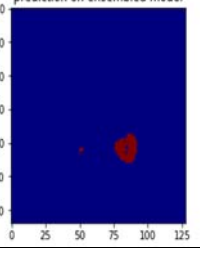
Input Image	Testing label	Ensemble Model Outcome	Ensemble Model	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
			0.9803	0.9615	0.9615	0.9615	0.9803	0.9803	0.9607
			0.9780	0.9753	0.9753	0.9753	0.9645	0.9726	0.9486
			0.9293	0.9359	0.9160	0.9192	0.9326	0.9230	0.8594
			0.9281	0.8849	0.9030	0.9207	0.9126	0.8943	0.8635
			Nan	Nan	Nan	Nan	Nan	Nan	Nan
			0.9622	0.9483	0.9573	0.9668	0.9622	0.9575	0.9624

Fig. 3. Results of Ensemble model with six top models used in Attention U-Net architecture

6. Conclusion

This study highlights segmentation of kidney with its tumor using Attention U-Net, FPN and LinkNet Architectures. Three Architectures used for segmentation is tested on the dataset for the KiTS Challenge 2019. For accurate segmentation of kidney and its tumors in a fast, accurate and automatic manner, highly reliable models are required. Three models can be used by specialists for improved diagnosis based on lower-level models that can be fine-tuned without GPUs using the pretrained models. It was particularly challenging because not all of the slices had both kidney and tumor tissues, resulting in additional computational demands. Pre-trained models were employed in experiments to simplify calculations and to save computation time. All the approaches in this study are evaluated with promising results for segmentation of kidneys and tumors based on semantics. With the addition of data augmentation or by including sagittal and coronal views of CT scans along with an axial view might achieve even better results. It is noted that ensembling the top six models using Attention U-Net Architecture is able to achieve high IoU scores (0.9566, 0.9386) and Dice Scores (0.9778, 0.9683) for kidney and tumor compared to FPN and LinkNet Architectures. The proposed approach is more generic and straightforward in comparison with sophisticated deep learning models with decent results currently available, can be used for segmentation of different organs and modalities. This process can also use different architectures for segmentation to improve the results.

Acknowledgment

This work was carried out using the equipment funded by the AICTE [Grant No. F.No.9-29/IDC/MODROB/Policy-1/2019-20].

References

- [1] Cairns, P.(2011). Renal Cell carcinoma, *Canc Biomarkers*, 9(1-6):461-73.
- [2] Rossi, S.H.; Klatte,T.; Usher-Smith,J.; Stewart,G.D.(2018).Epidemiology and screening for renal cancer, *World J. Urol.* 36 (9) 1341–1353, <http://dx.doi.org/10.1007/s00345-018-2286-7>, Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

- [3] Luana Batista da Cru ; Jose Denes Lima Araujo; Jonnison Lima Ferreira ; Joao Otavio Bandeira Diniz; Aristofanes Correa Silva; Joao Dallyson Sousa de Almeida; Anselmo Cardoso de Paiva; Marcelo Gattass. (2020). Kidney segmentation from computed tomography images using deep neural network, Computers in Biology and Medicine, volume 123, <https://doi.org/10.1016/j.combiomed.2020.103906>.
- [4] Nicholas Heller; Niranjana Sathianathan; Arveen Kalapar; Edward Walczak; Keenan Moore; Heather Kaluzniak; Joel Rosenberg; Paul Blake; Zachary Rengel; Makinna Oestreich; Joshua Dean; Michael Tradewell; Aneri Shah; Resha Tejpal; Zachary Edgerton; Matthew Peterson; Shane Abbas Raza; Subodh Regmi, Nikolaos Papanikolopoulos; Christopher Weight. (2019). The KiTS19 Challenge Data: 300 Kidney Tumor Cases with Clinical Context, CT Semantic Segmentations, and Surgical Outcomes, ArXiv abs/1904.00445 (2019).
- [5] Prashant Jadia. (2019). Kidney Tumor Segmentation Using Deep Learning, International Journal of Science and Research (IJSR), ISSN: 2319-7064, Volume 8, Issue 12, December 2019, DOI: 10.21275/ART20203792.
- [6] Haghighi, M.; Warfield, S.K.; Kurugol, S. (2018). Automatic renal segmentation in dcmri using convolutional neural networks, 2018 IEEE 15th International Symposium on Biomedical Imaging, pp. 1534–1537, <http://dx.doi.org/10.1109/ISBI.2018.8363865>.
- [7] Mredhula, L.; Dorairangaswamy, M.A. (2015). DETECTION AND CLASSIFICATION OF TUMORS IN CT IMAGES, Indian Journal of Computer Science and Engineering (IJCSSE), ISSN: 0976-5166, Vol. 6 No.2.
- [8] Chuanxia Wang; Yuting He; Xiaoming Qi; Ziteng Zhao; Guanyu Yang; Xiaomei Zhu; Shaobo Zhang; Jean-Louis Dillenseger; Jean-Louis Coatrieux. (2019). BiSC-UNet: A fine segmentation framework for kidney and renal tumor, 2019 Kidney Tumor Segmentation Challenge: KiTS19, Miccai 2019, Oct 2019, Shenzhen, China, 10.24926/548719.013, hal-02358682.
- [9] Seda Arslan Tuncer; Ahmet Alkan. (2019). Spinal Cord Based Kidney Segmentation Using Connected Component Labeling and K-Means Clustering Algorithm, International Information and Engineering Technology Association (IIETA), Vol. 36, No. 6, pp. 521-527, <https://doi.org/10.18280/ts.360607>.
- [10] Ozan Oktay; Jo Schlemper; Loic Le Folgoc; Matthew Lee; Matthias Heinrich; Kazunari Misawa; Kensaku Mori; Steven McDonagh; Nils Y Hammerla; Bernhard Kainz; Ben Glocker; Daniel Rueckert. (2018). Attention U-Net: Learning Where to Look for the Pancreas, 1st Conference on Medical Imaging with Deep Learning (MIDL), Amsterdam, The Netherlands, arXiv:1804.03999v3[cs.CV].
- [11] Lin, T.Y.; P. Dollár, R.; Girshick, K.; Hariharan, B.; Belongie, S. (2017). Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125.
- [12] Chaurasia, A.; Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation, in: 2017 IEEE Visual Communications and Image Processing (VCIP), IEEE, pp. 1–4.
- [13] Simonyan, K.; Zisserman, A. (2014). Very deep convolutional networks for largescale image recognition, arXiv preprint arXiv:1409.1556 (2014).
- [14] Kaiming He; Xiangyu Zhang; Shaoqing Ren; Jian Sun. (2016). Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.
- [15] Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. (2017). Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708.
- [16] Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861.
- [17] Tan, M.; Le, Q.V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks, arXiv preprint arXiv:1905.11946.
- [18] Jie Hu; Li Shen; Samuel Albanie; Gang Sun; Enhua Wu. (2019). Squeeze-and-Excitation Networks, arXiv:1709.01507v4.
- [19] Christian Szegedy; Sergey Ioffe; Vincent Vanhoucke; Alex Alemi. (2016). Inception-v4 Inception-ResNet and the Impact of Residual Connections on Learning, arXiv:1602.07261v2.
- [20] Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. (2015). Rethinking the inception architecture for computer vision. ArXiv preprint arXiv:1512.00567.
- [21] Jonathan Long; Evan Shelhamer; Trevor Darrell. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440.
- [22] Olaf Ronneberger; Philipp Fischer; Thomas Brox. (2015). U-net: Convolutional networks for biomedical image segmentation, In International Conference on Medical image computing and computer-assisted intervention, pages 234–241.

Authors Profile



Geethanjali.T.M., is currently serving as Assistant professor, Department of Information Science and Engineering at P.E.S. College of Engineering, Mandya, Karnataka, India. She has a teaching experience of 21 years. She is currently pursuing her PhD in Medical image processing and Computer vision. She has 04 research articles published in International Journals/Conferences.



Dr. Minavathi, is currently serving as Professor, Department of Computer Science and Engineering and Dean Research at P.E.S. College of Engineering, Mandya, Karnataka, India. She has about 31 years of experience in teaching. She is an active researcher and academician. Her areas of research include image processing, computer vision and machine learning. She has over 30+ research articles published in International Journals/Conferences.