An Efficient and Optimal Deep Learning Architecture using Custom U-Net and Mask R-CNN Models for Kidney Tumor Semantic Segmentation

Sitanaboina S L Parvathi[1]
Research Scholar, School of Computer Science and Engineering, Vellore Institute of Technology
VIT-AP University, Vijayawada, India

Harikiran Jonnadula[2]
Associate Professor, School of Computer Science and Engineering, Vellore Institute of Technology
VIT-AP University, Vijayawada, India

***Abstract*—Today, kidney medical imaging has become the backbone for health professionals in diagnosing kidney disease and determining its severity. Physicians commonly use Computerized Tomography (CT) and Magnetic Resonance Imaging (MRI) scan models to obtain kidney disease information. The significance and impact of kidney tumor analysis drew researchers to semantic segmentation of kidney tumors. Traditional image processing methodologies, in general, require more computational power and manual assistance to analyze kidney medical images for tumor segmentation. Deep Learning advances are enabling less computational and automated models for kidney medical image analysis and tumor lineation. Blobs (regions of interest) detection from medical images is gaining popularity in kidney disease diagnosis and is used widely in detecting tumors, glomeruli, and cell nuclei, among other things. Kidney Tumor segmentation is challenging compared to other segmentation models due to morphological diversity, object overlapping, intensity variance, and integrated noise. In this paper, It have proposed a kidney tumor semantic segmentation model based on CU-Net and Mask R-CNN to extract kidney tumor information from abdominal MR images. Initially, It trained the Custom U-Net architecture on abdominal MR images with kidney masks for kidney image segmentation. The Mask R-CNN model is then used to lineate tumors from kidney images. Experiments on abdominal MR images using Python image processing libraries revealed that the proposed deep learning architecture segmented the kidney images and lined up the tumors with high accuracy.**

Keywords—Kidney tumor (Blob) detection; custom U-Net; mask R-CNN; semantic segmentation; deep learning; medical image processing

I. Introduction

Medical imaging provides high resolution and coverage for the visualization of specific body organs. X-Ray, CT, MRI, and PET-CT scans are some frequently used medical imaging [1] technologies. In general, physicians will manually analyze the content of medical images to identify disease information. Image processing [2] and deep learning technologies [3] have been providing disease diagnosis models for over a decade, removing human errors in disease prediction. Deep learning for medical image analysis has attracted researchers and medical analysts because it requires less human intervention in data labeling and depth processing models than traditional image processing methods. Diseased regions (biomarkers) in a medical image differ in properties (i.e., the contrast in brightness) from their neighbor pixels and appear as blobs in nature. These biomarkers or blobs are the regions of interest (ROI) in medical image diagnosis, and they must get identified, segmented, classified, and labeled to predict disease. Image analysis models based on deep learning will assist in detecting biomarkers in medical images and provide spatial information such as location, shape, scale, inertia, and convexity. The Blob (or image ROI) detection [4] allows disease diagnosis in many instances of the medical image diagnosis such as brain tumors, kidney glomeruli, eye retina, breast lesions, and cell nuclei detection, among others. Medical image biomarkers detection models are becoming prominent applications for physicians in disease conformation, staging, and treatment planning.

Due to the importance of biomarkers detection in medical image analysis, many former researchers were focused on this topic and proposed various deep learning models for medical image blobs detection. Although many scholars have worked on medical image diagnosis, some prominent literature aided us in selecting the objectives and designing the proposed system using deep learning technologies. Parvathi et al. [5] integrated deep learning and image processing algorithms to execute the blob detection and classification operations on kidney 3D MRI images. They used the ECLAHE and IMBKM models to segment the blobs from the input images and later the deep learning IMBKM and EDCNN classifiers to classify the blobs into the selected disease categories. Xu et al. [6] created a hybrid model that used a deep U-Net model and hessian analysis to detect small blobs in 3D MRI images for kidney glomeruli detection. They designed a superset of blobs using the hessian analysis to distinguish the real-convex blobs from the noisy ones. Their custom deep learning model UH-Net integrated the hessian superset information and the U-Net pre-training knowledge to find the glomeruli through small blobs detection from the 3D kidney MRI images. Peng et al. [7] proposed a multi-scale blob detection model for automated stem cell segmentation from the underlying microscopic image set. The cell boundaries are delineated with high accuracy using blob and centerline detection.

In the literature survey, it went through many research articles as part of the medical image disease prediction and identified some research gaps, which are as follows: Because of the morphological diversity, segmenting the kidney object from the multi-organ medical image using shape and location information yields less accuracy. The traditional semantic segmentation approach, which may detect object categories in medical images, is insufficient because the medical image contains many instances of the same object type. Therefore, the instances must be categorized as it will.

Since kidney cancer became a major cause of kidney failure in people, our primary research goal was to detect kidney tumors from MRI images. To achieve this goal, it planned to segment the kidney objects from the MR first, and the lineation of the tumors from the kidneys is later. Image processing models (such as SIFT [8] and SURF [9]) and deep learning models (such as DNNs [10], CNNs [11], and Res-Net [12], among others) are the two different technologies used to segment the kidneys from CT images. Although each model has its pros and cons for kidney segmentation, it is interested in deep learning models because they are computationally cheaper and allow for high-level automation of the segmentation process. Unlike commonly performed object segmentation from images, kidney segmentation is a unique and challenging task, as shown in Fig. 1, because diseased kidneys segmentation has issues due to morphological diversity, object overlapping, intensity variance, and integrated noise.

Kidney tumor segmentation from MR images is a two-step process that includes kidney object (with tumor) segmentation and tumor object lineation. To accomplish the kidney tumor segmentation task and address segmentation issues (research gaps), in this paper, it proposed an efficient and optimal kidney tumor segmentation architecture using the custom U-Net and Mask R-CNN [13] deep learning models. The custom U-Net model is used first for kidney segmentation, and the Mask R-CNN model is used to lineate tumors from the segmented kidney images. The custom U-Net model is used first for kidney segmentation, and the Mask R-CNN model is used to lineate tumors from the segmented kidney images. To demonstrate the efficiency of the proposed kidney tumor segmentation model, a set of MR images collected from the TCGA-KIRC dataset and a python prototype is implemented to conduct the experiments.

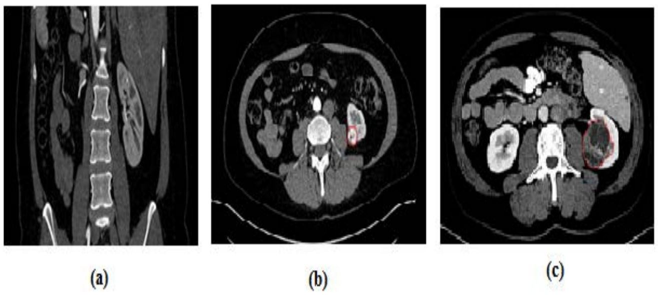


Fig. 1. Abdominal MR Scan Images. (a) Kidney with no Tumor; (b) Kidney with Mild Tumor; (c) Kidney with Moderate Tumor (Kidney Tumors Marked with Red Line in b and c).

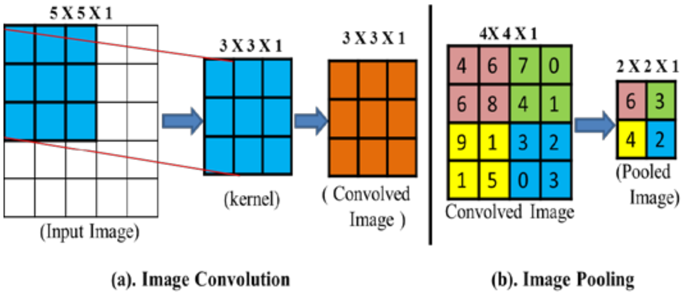
II. Related Work

In this section, it will discuss the Key technologies that have been used in our kidney tumor semantic segmentation processes like CNNs [14], U-Net [15] and Mask R-CNN [13].

A. CNNs

Recent advances in high-speed internet and mobile technology have resulted in a digital multimedia world with tons of images and videos. Computer vision is an emerging future domain, which is responsible for video and image manipulation as needed. Extracting useful information from an image is a complicated task because that needs to process a high volume of pixels. For over a decade, multi-layered deep learning models have made image processing easier than traditional methods. Because of their high accuracy and fully connected layers, Convolutional Neural Networks (CNNs) have proven to be the dominant deep learning model among the various deep learning models. As a descendant of Artificial Neural Networks (ANNs) [16], CNNs [14] will automatically train the feature maps from pixel arrays and identify the receptive fields through backpropagation using Key methods like convolution, pooling, and fully connected networks (FCN) [17].

1) Convolution: In general, the image is a collection of pixels, and these pixels will get converted into the respective intensity (RGB and grayscale) values for representation in a binary matrix model, which is feasible for manipulations. To manipulate the images with less computational overhead, they should be resized to a small size (down sampling) while retaining their receptive field (context) information. Convolution [18] is an affine transformation model in which the selected kernel matrix (filter) elements are iteratively multiplied against the input image matrix elements to generate the small-sized output convolved matrix. Fig. 2(a) represents the convolutional model with an input image (5x5x1) and the kernel (3x3x1) with a stride value of 2, and also the output convolved image (3x3x1). When it comes to the 3D images with RGB values, three color channels map with three different kernels for computations, and the final summation value is added with the bias to generate the convolved output image.



(a). Image Convolution

(b). Image Pooling

Fig. 2. Input Image Convolution and Pooling Model.

2) Pooling: In CNN, the convolution is followed it by a pooling [19] process, which mimics the behavior of the convolution process to reduce the convolved image spatial size at a high rate. Using either the max or average pooling method reduces computational overhead while learning and regulates over-fit issues. In Fig. 2(b), the average pooling method with a 2x2 filter and stride value of 2 is applied on a convolved image (4x4x1) to generate a pooled feature map (2x2x1). Like the convolution process, the pooling does not support the zero-padding at input feature map borders.

3) FCN: The FCN [17] is a feed-forward neural network model for multilayer perceptron that enables the backpropagation to learn the key features for classification across the epochs. The high-level features generated by the final convolution/pooling process are encoded using a single (flattened) vector and fed into the FCN. The FCN calculates the loss-entropy value for decision-making against classification uncertainty to reduce false positives in classification.

B. U-Net

Ronneberger [15] designed U-Net, as it knows deep convolutional network architecture, for the semantic segmentation of objects from images, with high speed and precision. Extraction of the ROI is frequently required task in biomedical images, particularly for organ segmentation, objects localization. An efficient deep learning-based semantic segmentation model, such as U-Net, is required to achieve this ROI extraction. Compared to other CNN architectures [20], the U-Net is the most adaptive for medical image segmentation because of additional benefits such as pixel-level segmentation, limited training data, end-to-end training, pixel padding support, de-convolutions, and elastic deformation.

The two sections of the U-Net architecture are the contracting path (encoder part) and the symmetric expanding path (decoder part). The encoder part of U-Net, like convolutional networks, is continuous with convolutions and pooling methods to make the context more precise and sharper. This encoder reduces the dimensionality of the input feature without losing context, allowing the process to complete with less computation. The encoder performs the 3x3 convolutions iteratively till the pivot point. And after each convolution process, the batch normalization and the activation function [21] (Rectified Linear Unit (ReLU)) are applied with pooling strides for down-sampling, which helps in the précised context making. In contrast, the U-Net has a decoder part with a transposed convolution mechanism on the other side, making the U-Net an end-to-end FCN model. The encoder output is up-sampled [22] by the decoder using convolutions, batch normalizations, and ReLU activations. The decoder is intended to return the output with precise localization using the up-sampling process and transposed convolution.

C. Mask R-CNN

As part of their research on AI, the Facebook AI Research Team (FAIR) introduced the instance segmentation framework called Mask R-CNN [10] as an extension to the Faster R-CNN [23] by adding the ROI segmentation masks. Compared to the other models, the Mask R-CNN is fast, simple, flexible, and accurate in instance segmentation is proven by the COCO - 2016 challenge. Mask R-CNN can segment the multiple instances precisely from the images, using image localization, object detection, and segmentation methods. Mask R-CNN architecture was designed by joining the Faster R-CNN with FCN model [24] for instance segmentation process.

Initially, a set of input images with different class objects are selected for instance segmentation. Deep CNN architectures [20] with convolution and pooling operations will extract the ROI bounding boxes from the input images. Unlike the Region-based CNN model, the Mask R-CNN had the ROI alignment phase, in which the exact spatial ROI volumes are identified from input images based on the input masks, using the pixel to the pixel alignment process. Mask R-CNN evaluates the ROIs from ROI-Pool and in parallel performs the target object detection to overcome the performance issues. Mask R-CNN evaluates the ROIs from ROI-Pool and in parallel performs the target object detection to overcome the performance issues. The bounding boxes are scaled using the Intersection over Union (IoU) metrics [25] after completing the ROI alignment process, and misinterpretations get eliminated using the feature matching threshold value. At this point, various class masks are applied to the coarse-grained bounding boxes to find the fine-grained segmentation. Finally, these fine-grained segmentations are precisely lineated and masked.

III. Kidney Tumors Segmentation Architecture using CU-Net and Mask R-CNN Models

In recent times, kidney tumor diagnosis from the medical images becomes a focusable research area due to its impact and importance in disease diagnosis and staging. The contribution of tumor detection is invaluable in cancer disease staging and treatment (especially in targeted therapy) planning. Researchers are interested in medical image analysis using deep learning models over traditional image processing techniques to reduce the computational (i.e., hardware) difficulties [26] in medical image processing. It is discussed in Section I that the kidney tumor diagnosis from MRI images faces several issues since this process differs slightly from the regular object segmentation and lineation process. To address the issues involved with the kidney tumor semantic segmentation process, It designed an efficient and optimal deep learning architecture (shown in Fig. 4) using the Custom U-Net [10] and Mask R-CNN models for kidney image segmentation and the tumor instance lineation from medical MR images. Kidney tumors can be segmented [27] from the medical MR images in two phases: Kidney(s) segmentation from MR images and tumors boundary lineation from kidney images.

A. Kidney Segmentation Phase

The process of extracting kidney images (with tumors) from MR scan images is known as kidney segmentation [28]. Later these extracted kidney images are used for tumor detection and its boundary lineation. For this, it selected a set of abdominal MR images and their associated ground truth values (masks) as a training dataset (shown in Fig. 3) for the experimental analysis. However, extracting the kidney context from the MR images is a complex process due to several reasons. Because of the cancer disease [29], the shape of the kidneys is inconsistent in nature and different from one other, which will make the network's training process difficult. Because the MR scan images are collected from different MR scanners, the intensity of the input images varies, which is incompatible with many deep learning networks. In addition to kidneys and other organs, noisy data is often present in MR scan images, making the object detection process more complex.

1) Custom U-Net: To solve these challenges in the kidney extraction process, it used the custom U-Net [10] model, a trained convolutional neural network that is suited for medical image segmentation. U-Net model [15] was selected over the other CNNs [21] because the U-Net supports the classification at pixel level and is suitable for the multi-class instance labeling if required. Due to the systematic hurdles [30] involved in data collection and processing, obtaining a dataset with tons of images and masks is impossible related to medical images. U-Net is a light-tight model because it can train efficient models with limited training datasets. Unlike trained nets [20] such as LeNet-5 and Dense DNN, the U-Net is free from dense layers and thus accepts input data with intensity variance.

It designed a custom U-Net model with additional features by extending the traditional U-Net to support kidney object segmentation with high accuracy. Our custom U-Net model is designed with a validation set to ensure test accuracy. By adjusting the hyper-parameters at validation time, the custom U-Net tunes the deep model to achieve high accuracy in test results. CU-Net was enhanced with data augmentation techniques such as image flips and others to double the size of the input dataset to generalize the training model. Dropout regularization functions it added to CU-Net during the training phase after each max-pooling operation to randomly replace neurons (pixels) with zero values and train the model with alternative neural networks to minimize overfitting.

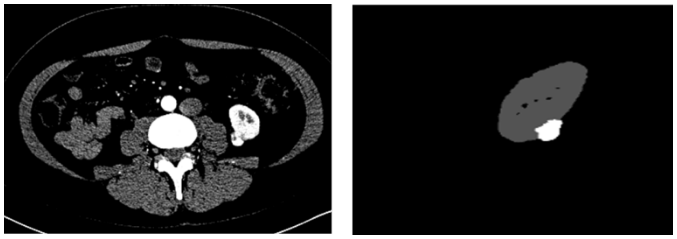


Fig. 3. An Abdominal MR Image (356 x 356) with Kidney Tumor (Left) and its Ground Truth Image for Training (Right).

2) Dataset model: A set of total $'n'$ MR scan abdomen images $\underset{\sim}{I} = \{I_1, I_2, I_3 \ldots I_n\}$ containing multiple organs with dimensions $D = [d_1, d_2]$, pixels $P = (p_1, p_2)$, are presented in a binary matrix model $M = [m_p \; x \; m_q]$. In our dataset, an image I_1 is having the greyscale pixel $(i_1(P))$ with its pixel value is represented as $0 \leq i_1(P) \leq 255$. Mask image set Ḿ is a set of total $'n'$ masks with information about kidneys with tumors segmentation, and it maps with their original MR images for training. The mask image set Ḿ $= \{M_1, M_2, M_3 \ldots M_n\}$ along with its label information L is presented as $M_k = \{(M_k, L) \in I_k \;\&\; L = 1\}$.

3) Training Custom U-Net: Once the data is defined and available in hand, the next immediate step in Custom U-Net is the data preparation. It classifies the data into a training set $\underset{\sim}{I}_\alpha$, validation set $\underset{\sim}{I}_\beta$, and test set $\underset{\sim}{I}_\gamma$. It used the validation set $\underset{\sim}{I}_\beta$ in the training phase to assess the model accuracy and to detect the overfitting problems [31] at the training phase itself. After partitioning the data set into train and test sets, a custom U-Net model with a contracting path and an expensive path was designed. The U-Net model is symmetric, with four layers of processing at each path. The 2D_convolution, max pooling, and dropout functions are implemented in the contracting path, whereas the transpose convolution, concatenate, and dropout functions are implemented in the expensive path, as shown in Algorithm 1. In U-Net, the contracting path's four layers are executed first, with convolution, max pooling, and dropout functions, and the results are passed to the next layer in the path. The convolution function increases the context of the input image $\underset{\sim}{I}_{in}$, which helps in target feature extraction, using the neurons (z), kernel (k_{m*n}), stride (s_{m*n}), activation function φ, an array of 4 channels $\mu = [1, 2, 4, 16]$, and padding p elements.

Down sampling [22] is a spatial dimensionality reduction method that reduces image height and width to make the image computationally feasible. After executing the convolution process twice, the resulted image volume is given as input to the max-pooling function to reduce the dimensionality of the image without losing the context. The ideal pool size (e_{m*n}) is selected and evaluated against the convolved image C_i to create the max pooled image Q_i. After max pooling, dropout functions with a frequency rate (f_{rate}) (0.0 - 1.0) are executed during the training phase. This function randomly sets the input pixels to zero, which helps to prevent values from dropping during the training phase and keeps the model from overfitting. This step completes a layer of the contracting path, and it will take four repeats to complete the entire contracting path. Soon after the contracting path completes, the convolution process will be repeated twice with the double neurons (μ) of the contracting path's last layer to build the connection (C_m) it has two paths.

Like the contracting path, the expensive path had four layers with transposed convolution, concatenation, dropout, and convolution functions but executed in backward direction. The Con2DTrans function performs the de-convolution process to reverse the convolution processes using the specified stride s_{m*n}, kernel k_{m*n}, and other attributes. The de-convolved image D_i is concatenated with its counterpart convolution image C_i to increase the dimensions. Dropout function can by the convolutions restores the image with equal dimensions of its counterpart contracting path layer. The same process will be repeated four times to conclude the expensive path execution. Finally, the single neuron and the 1x1 kernel-based convolution process will be executed to return the output image I_{out} with sharpened target context. In this manner, our proposed custom U-Net model is trained against the input image set to obtain the segmentation knowledge, which helps in validation and testing operations.

Algorithm-1: Custom U-Net Model Algorithm

```
Input: $I_{in}, k, s, p, z, \varphi, e_{m*n}$
Output: $I_{out}$
Method:
$cn_1, cn_2, cn_3, cn_4, \mu = [v_1, v_2, v_3, v_4]$
// contracting path
for i=1 to 4 do
  $C_i = con2D((z * \mu\,[i-1]), k_{m*n}, \varphi, p)(I_{in})$
  $C_i = con2D((z * \mu\,[i-1]), k_{m*n}, \varphi, p,)(C_i)$
  $cn_i = I_{in}$
  $Q_i = Max\_Pool(e_{m*n}, C_i)$
  $Q_i = D\_Out(f_{rate})(Q_i)$
  $I_{in} = Q_i$
end // for
// connected layers
  $C_m = con2D((z * \mu\,[3] * 2), k_{m*n}, \varphi, p)(I_{in})$
  $C_m = con2D((z * \mu\,[3] * 2), k_{m*n}, \varphi, p)(C_m)$
  $\mu = [v_4, v_3, v_2, v_1]$
// expensive path
for i=4 to 1 do
  $D_i = con2DTrans((z * \mu_{i-1}), k_{m*n}, s_{m*n}, \varphi, p)(C_m)$
  $D_i = concat(D_i, C_i)$
  $D_i = D\_Out(f_{rate})(D_i)$
  $D_i = con2D((z * \mu\,[i-1]), k_{m*n}, \varphi, p)(D_i)$
  $D_i = con2D((z * \mu\,[i-1]), k_{m*n}, \varphi, p,)(D_i)$
  $C_m = D_i$
end //for
  $I_{out} = con2D(1, k_{1*1}, \varphi, p,)(cn_1)$
return $I_{out}$
```

Regular classification models treat the validation set as an optional activity because it consumes more time for validations. But in our custom U-Net segmentation process, it generated the validation set $\underset{\sim}{\mathrm{I}}_{\beta}$ to monitor the model performance and hyper parameters tuning [32] according to the requirements. After the training process designed a segmentation model ω, the validation set $\underset{\sim}{\mathrm{I}}_{\beta}$ assesses the model performance at the training phase itself and fine-tunes the parameters through the backpropagation method if required. At the test phase, these fine-tuned models will assure the précised ROI segmentation. Due to the complexity involved in pixel-level processing, the medical image training may encounter the overfitting [31] problem, which arises when the trained segmentation model performance is specific and bounded to the training dataset only. In this case the trained model yields the best results on training data $\underset{\sim}{\mathrm{I}}_{\alpha}$ but it fails to segment the test data $\underset{\sim}{\mathrm{I}}_{\gamma}$. To overcome this over fitting issue in segmentation, it customized the U-Net to compare the trained model segmentation accuracy on both the training and validation datasets. The precision difference between both datasets will be compared against the over-fit threshold(δ) to confirm the overfit or the difference in performance $\mathbb{D}$ is shown below.

$$\mathbb{D} = \frac{1}{k}\sum_{i=1}^{k}\omega(\underset{\sim}{\mathrm{I}}_{\alpha}) - \omega(\underset{\sim}{\mathrm{I}}_{\beta})\ \{if\ \mathbb{D} \geq \delta\ than\ overfit\}$$

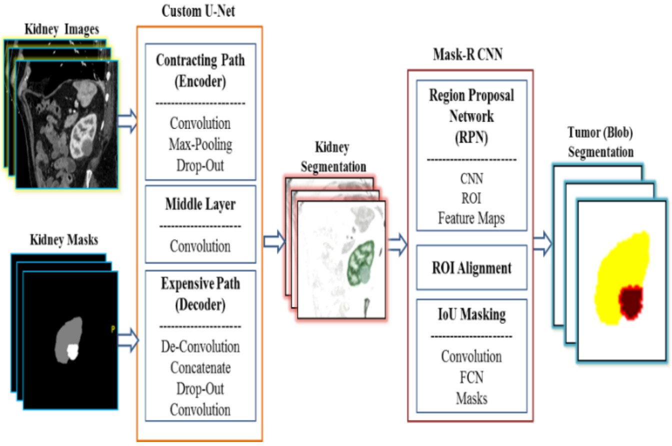


Fig. 4. Kidney Tumor Blob Instance Segmentation Architecture using CU-Net and Mask R-CNN Models.

B. Tumor Segmentation and Boundary Lineation

Because the kidney images (including tumors) it used to train the Custom U-Net based kidney segmentation model (shown in Fig. 4), the trained U-Net model returns segmented kidney images with tumors as a result. The tumor instances from the kidney images should be lineated and masked to find the tumor instance boundaries. Some prominent solutions include threshold-based segmentation, edge detection segmentation, feature clustering, bounding box, and ROI extraction. Among them, it selected Mask R-CNN [13], a fast, simple, and generalized ROI-based segmentation model for the target tumor object selection and precise segmentation (lineation) process. Stones, glomeruli, and tumors, etc. may appear as blobs in kidney imaging. Mask R-CNN detects a variety of blob objects on existing kidney images using the bounding boxes, and then the target tumors are identified using the shaded masks.

Convolution is used to extract feature maps from kidneys with tumor instances, and the Region Proposal Network (RPN) [23] is applied to the feature maps to generate bounding boxes for the target tumor blob. Based on the ROI volume, the bounding boxes are selected for further processing. ROI volume is evaluated using the Intersection over Union (IoU) approach, which compares the bounding boxes with the ground truth labels for ROI presence estimation. FCN has been used to detect the blob structures and mask them with the selected bounding boxes. Compared to the other segmentation models, this Mask R-CNN is lighter, faster, and reliable for pixel-level semantic segmentation.

IV. Experimental Analysis

To conduct the experiments on the proposed kidney tumor semantic segmentation architecture with CU-Net and Mask R-CNN models, it collected a set of 30 kidney MR images from the TCGA-KIRC dataset [33]. Images with 360x360 pixels and their corresponding masks (shown in Fig. 2) are extracted from these MR images for training and testing. The proposed Custom U-Net and Mask R-CNN based tumor segmentation architecture was implemented using the Keras-2.4.3 python interfaces on the TensorFlow-2.3.0 platform.

First, the MR image dataset is preprocessed and partitioned into train and test datasets, as explained in Section III. Train dataset images are transformed into two-dimensional binary arrays using the pixel data transformation methods for further processing. The images are now resized to 320x320 pixels for process compatibility with the custom U-net model. The contracting path and expensive path of the CU-Net are designed using convolutional and max-pooling methods respectively from the Kera's library. The proposed Custom U-Net model with the input layer and output layer uses the binary cross-entropy [34] as loss function and Adam's optimizer [35] for accuracy calculation with the prediction results. The dataset was augmented using image flips and other techniques to generate a set of synthetic data images derived from the core MR images. This step will increase the training data size by 2x more than the actual size to handle the overfitting issue in the classification process. Along with the data augmentation techniques, early stopping feature is also introduced to stop the training process at the appropriate time to avoid the over fit and under fit issues in training. To regularize the learning rate across multi epochs the learning rate reducing techniques also applied with CU-Net model. By specifying the epochs and the batch sizes for training process, the CU-Net model is trained on MR images to generate the efficient model for kidney image detection and segmentation.

The kidney segmentation binaries obtained from the Custom U-Net have been used as input to the Mask R-CNN model, which extracts tumor data. Mask R-CNN extracts objects from input images using bounding boxes and aligns the extracted objects using ROI information. Soon after the object localization using the bounding boxes alignment process, Mask R-CNN starts the pixels level comparison using the FCN to lineate the objects with specified shade masks. In Fig. 5, the kidney tumor segmentation results are shown along with the input images, masks, CU-Net kidney segments, and Mask R-CNN tumor lineation.

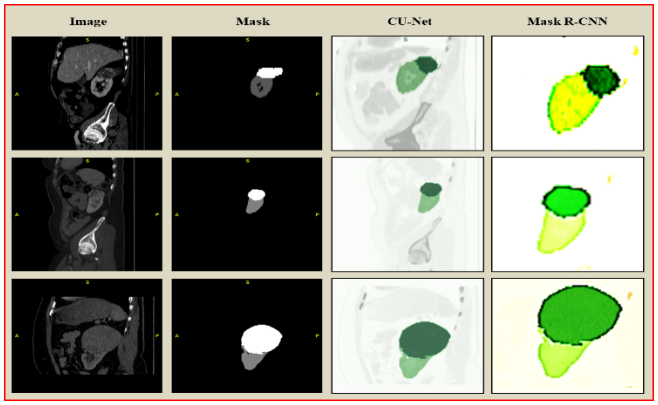


Fig. 5. Kidney Tumor Segmentation Results from the Experiments on Custom U-Net and Mask R-CNN Model.

Finally, the proposed model results have been evaluated using the loss and accuracy metrics from the prediction results on the test dataset. It adjusted the training and validation dataset proportions to test the accuracy and IoU metrics, and the results from the proposed architecture with CU-Net and Mask R-CNN are shown in Table I.

TABLE I. KIDNEY TUMOR SEGMENTATION ACCURACY AND IOU RESULTS

Data Partition	**IOU**	**Accuracy**
TD-70% and VD-6%	0.875	0.912
TD-75% and VD-8%	0.913	0.941
TD-65% and VD-15%	0.849	0.877
TD-60% and VD-20%	0.831	0.819

Fig. 6 depicts the generated validation results accuracy and loss value across multiple epochs for the proposed Custom U-Net model. The experimental results show that the proposed Custom U-Net and Mask R-CNN model is optimal, and it efficiently lineated blobs like kidney tumors with high lineation precision and segmentation accuracy.

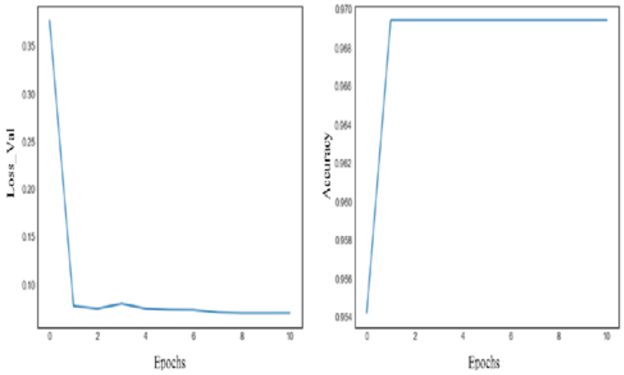


Fig. 6. Proposed Segmentation Model Results Accuracy and Loss Values across the Epochs.

V. DISCUSSION

For starters, this model allows for the use of global location and context at the same time. Second, it works with fewer training samples and outperforms other segmentation algorithms. Mask R-CNN outperforms all existing single-model entries on every task. Faster R-CNN is extremely efficient, with only a minor overhead added. Mask R-CNN can be easily adapted to other tasks.

VI. CONCLUSION

In this paper, it proposed the kidney tumor segmentation architecture with Custom U-Net and Mask R-CNN models. U-Net model is customized to overcome the kidney object segmentation issues like morphological diversity, object overlapping, intensity variance, and training overfit. Mask R-CNN is chosen to accurately lineate the tumor boundaries and segment (mask) the instances. The proposed architecture is used to train a set of MR scan images of kidney cancer, and the results are presented with the metrics IoU and accuracy. The experiments yielded high accuracy and IoU in kidney tumor segmentation and masking.

REFERENCES

[1] Krupinski EA, Jiang Y. Anniversary paper: evaluation of medical imaging systems. Med Phys. 2008 Feb;35(2):645-59. doi: 10.1118/1.2830376. PMID: 18383686.

[2] Angenent, S. & Pichon, Eric & Tannenbaum, Allen. (2006). Mathematical methods in medical image processing. Bulletin of the American Mathematical Society. 43. 365-396. 10.1090/S0273-0979-06-01104-9.

[3] Razzak M.I., Naz S., Zaib A. (2018) Deep Learning for Medical Image Processing: Overview, Challenges and the Future. Classification in BioApps. Lecture Notes in Computational Vision and Biomechanics, vol 26. Springer, Cham. https://doi.org/10.1007/978-3-319-65981-7_12.

[4] C. Duanggate, B. Uyyanonvara, S. Makhanov and S. A. Barman, "Enhanced support region for scale-space blob detection," in international conference on Robotics, Informatics and Intelligent control Technologies.

[5] Sitanaboina S L Parvathi, Dr. Harikiran Jonnadula, “Small Blob Detection and Classification in 3D MRI Human Kidney Images Using IMBKM and EDCNN Classifier”, Vol.12 No.5, 629-642, Turkish Journal of Computer and Mathematics Education, 2021.

[6] Xu, Yanzhe, Teresa Wu, F. Gao, J. Charlton and K. Bennett. “Improved small blob detection in 3D images using jointly constrained deep learning and Hessian analysis.” Scientific Reports 10 (2020). DOI:10.1038/ s41598-019-57223-y.

[7] Peng H, Zhou X, Li F, Xia X, Wong ST, “Integrating Multi-Scale Blob/Curvilinear Detector Techniques and Multi-Level Sets for Automated Segmentation of Stem Cell Images”. Proc IEEE Int Symp Biomed Imaging. 2009 Summer; 2009:1362-1365. doi: 10.1109/ISBI.2009.5193318.

[8] LoIt, D. G. “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision (2004).

[9] Bay, H., Ess, A., Tuytelaars, T. & Van Gool, L. “Speeded-Up Robust Features (SURF)”, Computer Vision and Image Understanding - 2008.

[10] Carneiro G., Zheng Y., Xing F., Yang L. (2017) Review of Deep Learning Methods in Mammography, Cardiovascular, and Microscopy Image Analysis. In: Lu L., Zheng Y., Carneiro G., Yang L. (eds) Deep Learning and Convolutional Neural Networks for Medical Image Computing. Advances in Computer Vision and Pattern Recognition. Springer, Cham. https://doi.org/10.1007/978-3-319-42999-1_2.

[11] Mortazi A., Bagci U. (2018) Automatically Designing CNN Architectures for Medical Image Segmentation. In: Shi Y., Suk HI., Liu M. (eds) Machine Learning in Medical Imaging. MLMI 2018. Lecture Notes in Computer Science, vol 11046. Springer, Cham. https://doi.org/10.1007/978-3-030-00919-9_12.

[12] Zhang Q., Cui Z., Niu X., Geng S., Qiao Y. (2017) Image Segmentation with Pyramid Dilated Convolution Based on ResNet and U-Net. In: Liu D., Xie S., Li Y., Zhao D., El-Alfy ES. (eds) Neural Information Processing. ICONIP 2017. Lecture Notes in Computer Science, vol 10635. Springer, Cham. https://doi.org/10.1007/978-3-319-70096-0_38

[13] He, Kaiming & Gkioxari, Georgia & Dollar, Piotr & Girshick, Ross. (2017). Mask R-CNN. 2980-2988. 10.1109/ICCV.2017.322.

[14] A. A. M. Al-Saffar, H. Tao and M. A. Talab, "Review of deep convolution neural network in image classification," 2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET), 2017, pp. 26-31, doi: 10.1109/ICRAMET.2017.8253139.

[15] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation.

[16] Filippo Amato, Alberto López, Eladia María Peña-Méndez, Petr Vaňhara, Aleš Hampl, Josef Havel, Artificial neural networks in medical diagnosis, Journal of Applied Biomedicine, Volume 11, Issue 2, 2013, Pages 47-58, ISSN 1214-021X, https://doi.org/10.2478/v10136-012-0031-x.

[17] E. Shelhamer, J. Long and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 4, pp. 640-651, 1 April 2017, doi: 10.1109/TPAMI.2016.2572683.

[18] Minsik Cho and Daniel Brand, “MEC: memory-efficient convolution for deep neural network”. In Proceedings of the 34th International Conference on Machine Learning - Volume 70 (ICML'17). JMLR.org, 815–824. . 2017.

[19] V. Christlein, L. Spranger, M. Seuret, A. Nicolaou, P. Král and A. Maier, "Deep Generalized Max Pooling," 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 1090-1096, doi: 10.1109/ICDAR.2019.00177.

[20] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8

[21] Nwankpa, Chigozie & Ijomah, Winifred & Gachagan, Anthony & Marshall, Stephen. (2020). Activation Functions: Comparison of trends in Practice and Research for Deep Learning.

[22] A. Youssef, "Analysis and comparison of various image downsampling and upsampling methods," Proceedings DCC '98 Data Compression Conference (Cat. No.98TB100225), 1998, pp. 583-, doi: 10.1109/DCC.1998.672325.

[23] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.

[25] Rahman, Md & Wang, Yang. (2016). Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation. 10072. 234-244. 10.1007/978-3-319-50835-1_22.

[26] Scholl, Ingrid & Aach, Til & Deserno, Thomas & Kuhlen, Torsten. (2011). Challenges of medical image processing. Computer Science - R&D. 26. 5-13. 10.1007/s00450-010-0146-9.

[27] Daza, Laura & Gomez, Catalina & Arbelaez, Pablo. (2019). Semantic Segmentation of Kidney Tumor using Convolutional Neural Networks. 10.24926/548719.077.

[28] N. Goceri and E. Goceri, "A Neural Network Based Kidney Segmentation from MR Images," 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), 2015, pp. 1195-1198, doi: 10.1109/ICMLA.2015.229.

[29] Nicholas J Vogelzang, Walter M Stadler, “Kidney cancer”, The Lancet, Volume 352, Issue 9141, 1998, Pages 1691-1696, ISSN 0140-6736, https://doi.org/10.1016/S0140-6736(98)01041-1.

[30] Kohli MD, Summers RM, Geis JR, “Medical Image Data and Datasets in the Era of Machine Learning”,Whitepaper from the 2016 C-MIMI Meeting Dataset Session. J Digit Imaging. 2017 Aug-30 pp:392-399. doi: 10.1007/s10278-017-9976-3.

[31] Horwath, J.P., Zakharov, D.N., Mégret, R. et al. Understanding important features of deep learning models for segmentation of high-resolution transmission electron microscopy images. npj Comput Mater 6, 108 (2020). https://doi.org/10.1038/s41524-020-00363-x.

[32] Schratz, Patrick & Muenchow, Jannes & Iturritxa, Eugenia & Richter, Jakob & Brenning, Alexander. (2018). Performance evaluation and hyperparameter tuning of statistical and machine-learning models using spatial data.

[33] Akin, O., Elnajjar, P., Heller, M., Jarosz, R., Erickson, B. J., Kirk, Filippini, J. (2016). Radiology Data from The Cancer Genome Atlas Kidney Renal Clear Cell Carcinoma [TCGA- KIRC] collection. The Cancer Imaging Archive. http://doi.org/10.7937/K9/TCIA.2016. V6PBVTDR.

[34] Zhang Z, Sabuncu M. Generalized cross entropy loss for training deep neural networks with noisy labels, In Advances in neural information processing systems 2018 (pp. 8778-8788).

[35] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” International Conference on Learning Representations (ICLR 2015), 2015.