Light-weight U-NET structure inducted in Kidney Image Segmentation for anomalies analysis

Manahil Shaikh[1], Bisma Khalid[1], Javaria Latif[1], Uzair Iqbal[2*]

[1]School of Computing, National University of Computing and Emerging Sciences, Islamabad, Pakistan

[2]Department of Software Engineering, Faculty of Computer Science and Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia

E-mail: uzairiqbal@um.edu.my

Abstract. Chronic kidney disease (CKD) refers to any long-term condition which deteriorates the functionality of the kidney in body waste filtration. The disease could be treated if diagnosed timely. Concerning image segmentation of chronic kidney disease, the increased computation cost of models due to varied shapes and sizes of tumors and position of stones, and varied range of image intensity of different forms of CKD makes it a challenging task to accurately segment the disease and kidney from an image Keeping this problem domain in view, this paper proposes a lightweight UNET based architecture for segmentation of chronic kidney disease using the concepts of machine learning models to trace the infected areas and create a base for kidney image analysis. This paper compares two different architectural changes to the UNET model to decrease the training time and training epochs. Furthermore, this paper uses feature upscaling in pre-processing to see the impact on the computational cost. Moreover, our technique includes promising opportunities as it can further be modified into a real-time device. Also, it can play an important role in a patient's health care and diagnosis.

1. Introduction

In medical terminology, chronic kidney disease is a generalized term describing a condition that degenerates a kidney and its functionality if left untreated for a long time. It is generally categorized into stones, tumors, and kidney cancers. In 2017, Chronic Kidney disease resulted in 1.2 million deaths and was the 12th leading cause of death worldwide. According to the research carried out in 2021 [1],15 percent of US adults are estimated to have chronic kidney disease. Due to the nature of the disease being asymptomatic, 9 of 10 adults do not know they have Chronic kidney disease and 2 of 5 adults with severe Chronic kidney disease do not know they have Chronic kidney disease. Patients with Chronic kidney disease are said to be at a higher risk of cardiovascular diseases due to anemic conditions and fluctuating blood pressure which eventually leads to poor quality of life and eventual death [2]

Hence for patients with Chronic kidney disease, the primary focus is on the early detection and prevention of the disease. The universally accepted method to test for the decrease in the functionality of the disease is the Glomerular filtration rate(GFR)

[2] [3]. GFR is a simple blood test that measures the waste levels from the digestion of protein in your body. The GFR levels are then used to diagnose the health of the kidney functionality [4].

The GFR test is a complex procedure that is generally considered impractical by health care officials it results in calculations of GFR levels using formulations which further may be prone to human error and incorrect diagnosis which creates the need for CT scans of the kidney for analyzing the kidney health using shape and structure of the kidney. The use of CT scans created a domain for image segmentation tasks to automate the process for the detection and diagnosis of chronic kidney disease. While keeping medical image segmentation in mind, the accuracy of the prediction of the disease plays an important role in correct diagnosis. The image segmentation models come with the inability to segment the disease correctly due to the varied shapes and sizes and high computational cost. The proposed approach, therefore, works on solving the following problems:

- The increased computation cost of the models to segment the disease
- Inaccurate segmentation results due to varied shapes and sizes of the tumor

However, we will propose;

(i) A lightweight U-Net-based architecture for the segmentation of chronic kidney disease using preprocessing techniques.

(ii) Also, we will compare our results with different machine-learning approaches.

(iii) Moreover, the preprocessing techniques include the feature upscaling schemes for reducing the computational cost.

2. Related Work

In [5], the authors use deep semantic segmentation learning models with a pro- posed training scheme to achieve precise and accurate segmentation outcomes of kidneys. The proposed training plan comprises two sectioning stages. In the primary stage, the first picture has been resampled and trimmed into a solitary locale covering both kidneys. This edited area is additionally cut into 3D shapes for the principal division organization. The subsequent stage gets the info in view of the main stage and unique picture, which the subsequent division organization could use for fine kidney segmentation and kidney stone recognition.

In [10], the authors proposed a boundary distance regression and pixel classification to segment kidneys automatically. A classification model was used as a starting point to obtain features from ultrasound images, the boundary dis- stance regression network learns kidney boundaries modeled as distance maps of the kidney boundaries, and the kidney boundary distance maps are finally used as input to the kidney pixel-wise classification network to generate kidney segmentation mask. A transfer learning strategy was adopted to extract relevant features from the images by using VGG16.

In [6], the authors present a multi-scale supervised 3D U-Net, MSS U-Net to segment kidneys and kidney tumors from CT images. The potential of the 3D U-Net architecture through deep supervision and exponential logarithmic loss was explored. Following the fundamental construction of 3D U-Net, the organization executes decoder and encoder components. The encoder layers are used to gain highlight portrayals from the info information. The decoder is then utilized for recovering voxel areas and for deciding their classifications in view of the semantic data extracted from the encoder.

This paper [7] proposes a transformer-based 3D image segmentation architecture, UNETR. This model employs the use of a stack of transformers and self-attention modules as encoders and deconvolution layers as decoders. The self-attention modules help architecture learning as it is able to learn weighted sums. At the bottleneck of the encoder path, a deconvolution layer is applied to the transformed feature map which increases image resolution by 2X. Each encoder is connected to the respective decoder by skip connections which combines the multi-resolution image data from encoders to decoders for efficient pixel-wise segmentation and to prevent the

loss of spatial information. The final 1x1 convolution layer with softmax activation function is used to generate pixel vise segmentation of the 3D image.

This paper [8] proposes an advanced segmentation technique, CR-UNet, to extract, encode and adaptively integrate multiple layers of correlated features. The basic architecture includes innovative Channel-Region (CR) which is a feature learning model, based on the UNet model, to learn the interrelation amongst the various levels of features. The contributions include two parts; channel-level feature learning and regional association attention mechanism. Channel seman- tic attention mechanism assign weights to various channel features in the same nodes by a 3D convolution layer with a kernel size of 1x1x1. In addition, the re-regional-association feature learning module applies weights to different points under the same channel. The convolution operation with 1x1x1 kernel size is applied to the output from UNet and then reshaped where each row corresponds to a certain image region.

This paper [9] proposes a C-CAM (Casual-Class Activation Mapping) technique for weakly supervised semantic segmentation (WSSS) images. WSSS images use weak labels that include an image-level label, point, scribble, and bounding box. The C-CAM approach includes two cause-effect chains, the category-causality chain, and the anatomy-causality chain. Wherein, category-causality chain is proposed to get through the problem of vague boundaries, it influences the classified categories with the disturbing context cofounder. Anatomy-causality chain is proposed to alleviate the co-occurrence problem; it includes the disturbance by anatomical structure on the segmentation shape. Moreover, the global sampling module is first used to generate the salient maps for each class and coarse segmentation masks. Then those masks and maps are forwarded to the category-causality chain as input. Where it is then passed through two con- convolutional layers to project the coarse segmentation masks and salient maps into the same space. After that, a category-aware attention vector with two con- volution operations and an image-specific category causality map with a downsampling operation is implemented to concatenate the results with CNN features. Furthermore, the anatomy-causality chain is designed as the 1/0 indicator where the anatomy-causality maps are calculated by downsampling and multiplied with the abdominal scans to get the resultant saliency maps. Finally, the pseudo-segmentation masks are calculated that are then passed through a U-Net model for segmentation.

In order to detect eye diseases, this study [11] suggests an interactive method for segmenting blood vessels in retinal fundus images. The technique makes use of the DRIVE dataset and the Canny edge detection algorithm. To achieve the desired vessel segmentation, the method involves altering the Canny edge detection settings via a graphical user interface (GUI). The RGB image is split into three channels (R, G, and B) and the green channel is subjected to the CLAHE technique in order to improve contrast. For noise reduction, Gaussian filters are employed. To improve edge detection, non-maxima suppression, and double thresholding are used. By fusing a binary mask and the Canny picture, the fundus contour is recovered. The method makes interactive adjustments to filter sizes and threshold values to overcome the drawbacks of CLAHE and Canny edge detection. When weak edges are difficult to detect, a manual mode for specifying edges is also available. In order to fine-tune the segmentation results, the research displays competitive performance and interactive control. The suggested approach might support computer-assisted ophthalmology and help with the detection and tracking of various eye conditions.

This study [12] introduces NEBCA, a brand-new clustering technique for segmenting MR images of Parkinson's disease (PD). NEBCA solves the issues of uncertainty and grayscale representation in medical imaging by utilizing neutrosophic set theory and the HSV color scheme. In order to improve contrast and enable better tissue distinction, the algorithm transforms grayscale images into color images. In comparison to other clustering algorithms, NEBCA applies clustering to the color images, improving segmentation accuracy, sensitivity, specificity, and dice similarity coefficient. The efficiency of NEBCA is demonstrated by the experimental

analysis performed on PD MR images. The suggested algorithm may help with the early identification and diagnosis of PD, which would enhance patient outcomes. NEBCA's modest computational complexity makes it useful for applications in the real world. Overall, NEBCA pro- vides a viable method for segmenting PD MR images and can be used in other medical imaging fields. Table 1 highlighted the detail literature review

Table 1: Analysis of Related Work

Reference	**Dataset**	**Data Process/ Data Collection**	**Accuracy**	**Pipeline**	**Weakness**
(Hatamizadeh et al., 2022)	BTCV (CT) MSD (MRI/CT)	The dataset is publicly available from MICCAI conference challenges	BTCV: 0.899 dice-coefficient Standard measure: 0.85 dice coefficient	Transformers and attentional network as encoders and deconvolution network as decoders combined with skip connections	High training times and training parameter
(Liu et al., 2021)	Kits19	Dataset is publicly available on GitHub	0.910 dice-coefficient	nnUnet as encoder and decoder, with the channel semantic attention mechanism and regional association model in between	High testing time and large memory requirement
(Chen, Tian, and Zhu, 2022)	Pro MRI ACDC CHAOS	Pro MRI is made of three subsets from PROMISE12, ISBI 2013, and in-house data ACDC is available on 2017 Automatic Cardiac Diagnosis Challenge (ACDC) CHAOS is public dataset from the challenge of CHAOS	Dice similarity coefficient on three datasets: 77.26%, 80.34%, and 78.15%	Salient Maps and segmentation masks from global sampling to category-causality chain as two conv layers. Anatomy-causality chain and finally U-Net model	Setting the precise threshold value for better accuracy. Also, it is difficult to segment crucial shapes.
(N. Heller, 2021)	Kits19	Dataset is publicly available on GitHub	The 1st place winner scored 0.974 Kidney Dice and 0.851 Tumor Dice making up 0.912 composite scores	3 architectures were implemented: plain 3D-UNet, residual 3D-UNet, and pre-activation residual 3D-UNet	High testing and training time.
(Shi Yin, 2019)	289 Ultrasound Images	Data was collected for routine clinical care	Proposed approach scored a 0.9304 dice coefficient	Transfer learning model to boundary distance regression network followed by a distance loss function, pixel-wise detection, and finally softmax loss function.	Requires large memory

3. PROPOSED METHODOLOGY

The proposed approach to this research work is divided into two parts namely; data preprocessing and model training and evaluation. For deployment, the website interface is created as a diagnostic tool for doctors and patients. Figure 1 highlight the proposed methodology.

The research is based on multiple experiments. Experiments are performed on complete data and also on randomly sampled parts of the dataset. Random sampling was done by generating

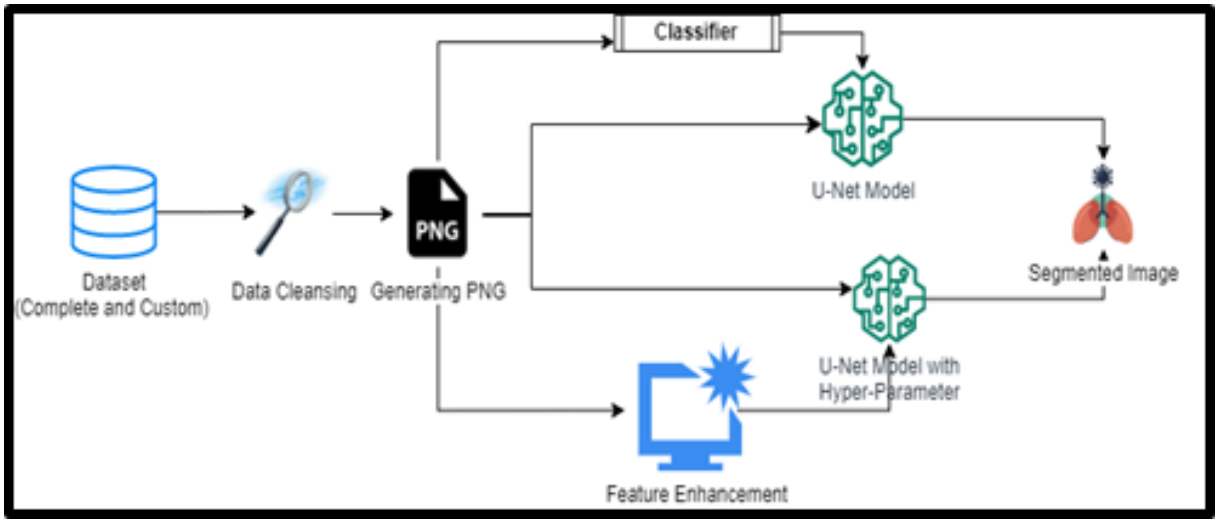


Figure 1: Proposed Method

100 random numbers and using those numbers as the patient is to be included in our custom dataset. For the first part, hyper-parameters were added and tested in the U-NET model such as increasing depth and adding dropout layers. Furthermore, in another experiment, a retrained classifier (VGG-16 AND Resnet505050) was used to first classify the images with ground truth segmentations and images without ground truth segmentation. As a CT-Scan is scanned from different angles to create a 3D image, the dataset contains some images having no presence of kidney or kidney disease. A classifier is used to exclude the images with no segmentations. For training the classifier, the dataset is created manually by separating the segmented and non-segmented images. After classification, the images containing kidney or kidney disease are passed to the U-NET model for segmentation. The results of both approaches are compared. Both these experiments are performed with basic image preprocessing such as data cleaning, image resizing, etc.

Lastly, a feature upscaling technique will be used as preprocessing to upscale similar features between ground truth segmentation and the input image. Then, the refined image will be passed to the U-NET model for segmentation

4. EXPERIMENTAL SCHEME

4.1. Model training using different hyper-parameters

The research is based on 3 sets of experiments. Experiments 1 and 2 include a slight modification in hyper-parameters of UNET whereas experiment 3 uses a classification-based approach leading to Chronic kidney disease segmentation. The results of each experiment are compared with the standard UNET model as proposed in this paper. [13].The details of the experiments are discussed below.

The data set used for this project is obtained from the kidney tumor segmentation challenge in 2019 (KITS19). The data set consists of 299 folders of which 209 folders include training and validation data set and the rest are for testing. Each folder in the data set is represented with a case of the patient and each folder contains two sub-folders for DI-COM images of CT-scan and its respective segmentation. In the data pre-processing step, the DI-COM image slices are each converted to PNG files and stored in the same organization manner as the initial files. The images were scaled to 256x256 dimensions. Experiments 1 and 2 are performed on complete data and also on randomly sampled parts of the data set. Random sampling was done by generating

100 random numbers and using those numbers as the patient is to be included in our custom data set.

For evaluation and analysis of the architecture's performance, the training time of each epoch is compared along with the dice coefficient and intersection over union(IOU). Here, the training time refers to the number of epochs at which the model training has stopped or shows no improvement in loss. The batch size and image size are kept constant for all experiments. All the experiments were performed using GPU P100 on Kaggle.

Scenario 1: UNET with Dropout layers

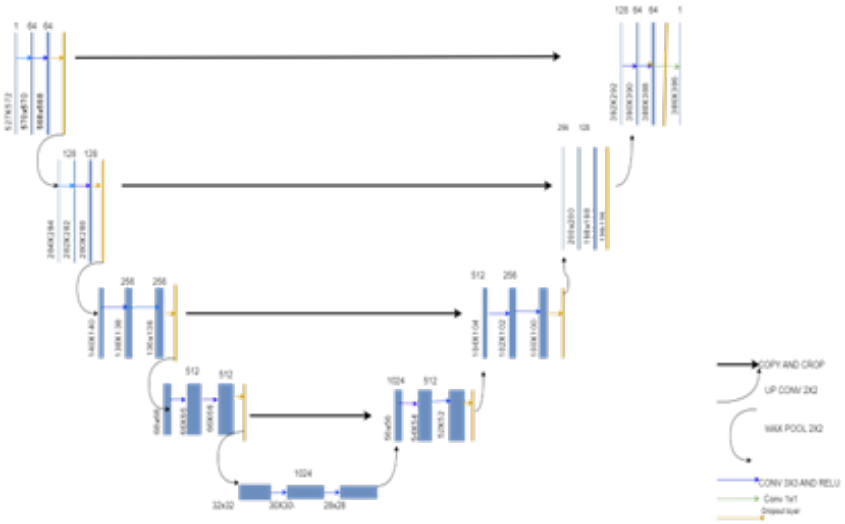


Figure 2:
U-Net architecture with dropout layers

Experiment Using full dataset: Figure2. shows the architecture used in this experiment. For this experiment, the encoders and decoders of the UNET model were modified and a dropout layer was added in each encoder and subsequent decoder.The dropout layers set the non-significant inputs to a layer of 0 randomly with the frequency of rate. The significant inputs are scaled up by a factor of 1/ (1 - rate). This prevents the overfitting of the model and also reduces the time a model needs to reach the maximum accuracy [14]. When trained on the full dataset, the training time of each epoch is compared. Here, the training time refers to the number of epochs at which the model training has stopped or shows no improvement in loss. The training time of UNET with dropouts is less than the time for UNET for dropout rate of 0.25,0.70 and 0.80 with dice coefficients 0.969,0.956.0.933 respectively. For a dropout rate of 0.10, the training time increases but archives a dice coefficient of 0.978. For dropout rates of 0.05 and 0.50, the decrease in time is insignificant. Hence, as the dice coefficient also decreases with a decrease in time, the second part of the experiment was performed as continued in experiment 2. The table below summarizes the results of experiment 1.

Table 2: Experiment 1 Results on Public Dataset

Model	Epochs	Dice Coefficient	IOU
UNET	9	0.977	0.955
Dropout Unet (rate = 0.05)	8	0.975	0.951
Dropout Unet (rate = 0.10)	13	0.978	0.957
Dropout Unet (rate = 0.25)	6	0.969	0.941
Dropout Unet (rate =.50)	8	0.969	0.94
Dropout Unet (rate =0.70)	7	0.956	0.915
Dropout Unet (rate =0.80)	5	0.933	0.875

Experiment Using Custom dataset: When trained on a custom data-set (with 100 randomly selected patient ids), the results for dropout UNET performed to be better than for UNET in terms of training time and dice coefficients in some cases. For dropout rates of 0.05, 0.25, and 0.70, the training time is less than that of UNET. The dice coefficient for dropout rates 0.05,0.25 and 0.70 are 0.974, 0.969, and 0.963 respectively. For dropout rates 0.10,0.50 and 0.80, there is no significant difference in training time. The dice coefficients for these rates are 0.977, 0.974, and 0.96 respectively. The table summarizes the results of the scenario below.

Table 3: Scenario 1 Results on Public Dataset

Model	Epochs	Dice Coefficient	IOU
UNET	12	0.97	0.941
Dropout Unet (rate = 0.05)	7	0.974	0.949
Dropout Unet (rate = 0.10)	10	0.977	0.954
Dropout Unet (rate = 0.25)	7	0.969	0.94
Dropout Unet (rate =.50)	12	0.974	0.95
Dropout Unet (rate =0.70)	9	0.963	0.929
Dropout Unet (rate =0.80)	13	0.96	0.923

Scenario 2: Training UNET with increased depth of the model

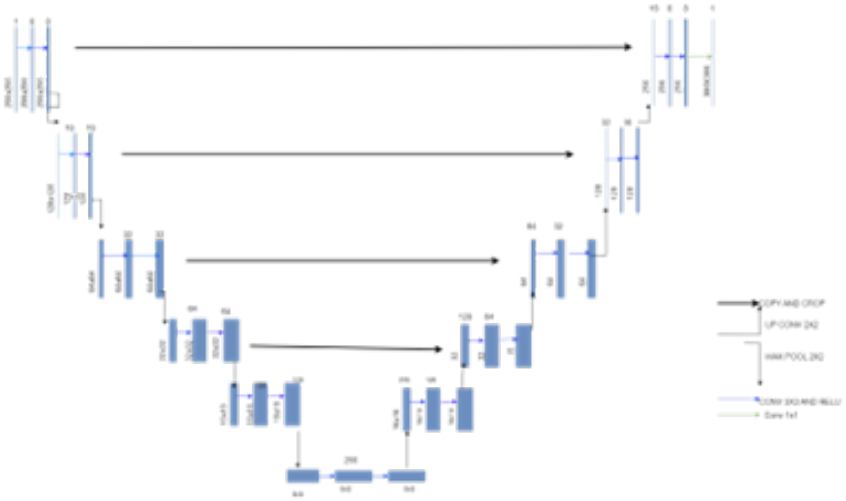


Figure 3: U-NET architecture with increased depth and reduced filters

Figure 3 shows the architecture used in this experiment UNET model was modified to include 5 encoders and 5 decoders instead of the basic UNET model which includes 4 encoders and 4 decoders. Figure 3. shows the architecture used for this experiment. Keeping the increased depth in mind, the number of filters in each encoder and decoder was decreased to half that of state-of-the-art UNET. On a full dataset, the proposed model took about 4 epochs to reach the dice coefficient of 0.971. On a custom dataset, the proposed model took about 7 epochs to reach the dice coefficient of 0.975. The results are shown below.

Table 4: Increased depth model results

Model	**Dataset**	**Epochs**	**Dice Coefficient**	**IOU**
ID-UNET	Public dataset	4	0.971	0.943
ID-UNET	Custom Dataset	7	0.975	0.952

Scenario 3: Pre-trained classification and then training with UNET

For this experiment, two pre-trained classifiers, VGG16 and Resnet50, were used to separate the segmented and non-segmented images. The drawback of this approach was that the classifiers have high computational costs and therefore require high computational resources. Also, the validation accuracy of training of vgg16 and ResNet was achieved to be 86 percent and 84 percent respectively. Hence, some images were incorrectly classified which affected the results of UNET and dropout U-Net as well. The classifiers were trained for 15 epochs each. The tables below show the results.

Table 5: Classification with VGG 16, segmenting with UNET

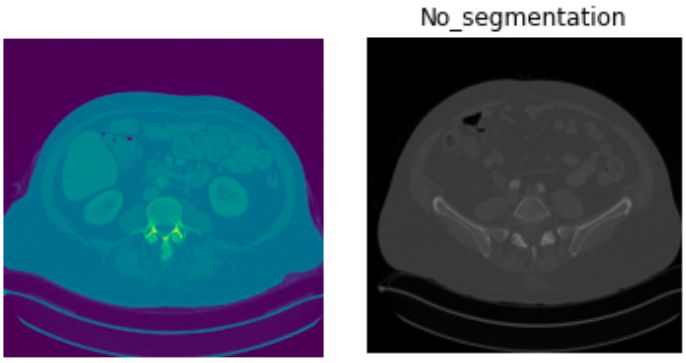
Epochs	**Training Accuracy**	**Training Loss**	**Learning Rates**	**Validation Accuracy**	**Validation Loss**
9	0.932	0.162	1.00E-05	0.824	0.648
10	0.977	0.057	1.00E-05	0.844	0.531
11	0.986	0.0356	1.00E-05	0.841	0.458
12	0.989	0.027	1.00E-05	0.855	0.377
13	0.992	0.018	1.00E-05	0.850	0.540
14	0.994	0.017	1.00E-05	0.852	0.535
15	0.998	0.006	1.00E-07	0.861	0.457

Table 6: Classification with RESNET50, segmenting with UNET

Epochs	**Training Accuracy**	**Training Loss**	**Learning Rates**	**Validation Accuracy**	**Validation Loss**
10	0.981	0.048	1.00E-05	0.854	0.589
11	0.988	0.031	1.00E-05	0.830	1.289
12	0.991	0.024	1.00E-05	0.838	0.734
13	0.997	0.011	1.00E-07	0.845	1.00

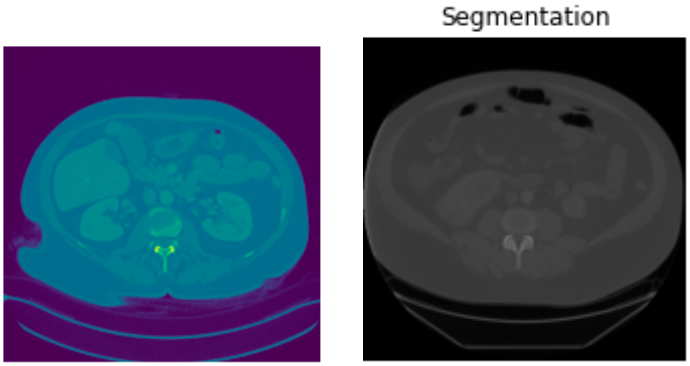
4.2. Feature selection on Input image for pre-processing

By definition, Feature selection in image processing refers to the process of selecting a subset of features or descriptors that are most relevant to a particular image analysis task. These features are typically numerical representations of image properties, such as texture, shape, or color,



(a) (b)

Figure 4: Classification with VGG16 with U-Net segmentation (a) Raw image (b) Final predicted image



(a) (b)

Figure 5: Classification with ResNet50 with U-Net segmentation (a) Raw image (b) Final predicted image

and they are used to characterize or classify images. The selection of relevant features is an important step in image processing because it reduces the dimensionality of the problem and improves the efficiency and accuracy of subsequent image analysis tasks.

Scenario 1: Pre-processing using Canny Edge Detection This paper uses the Canny edge detection [15] algorithm for edge detection. In image processing, edges refer to the discontinuities in the intensity values of adjacent pixels in an image. These changes could occur due to different factors such as object boundaries, texture changes, or lighting variations. Edges are typically represented as a set of connected points that form a curve or a line, which separates different regions in an image. The following equations explain the working of the canny edge detection algorithm The following equations explain the working of canny edge detection algorithm:

Step 1 : Applying Gaussian smoothing [16] on the image

The image is convolved with Gaussian filter to apply noise reduction and smoothing.The equation for Gaussian filter is given below.

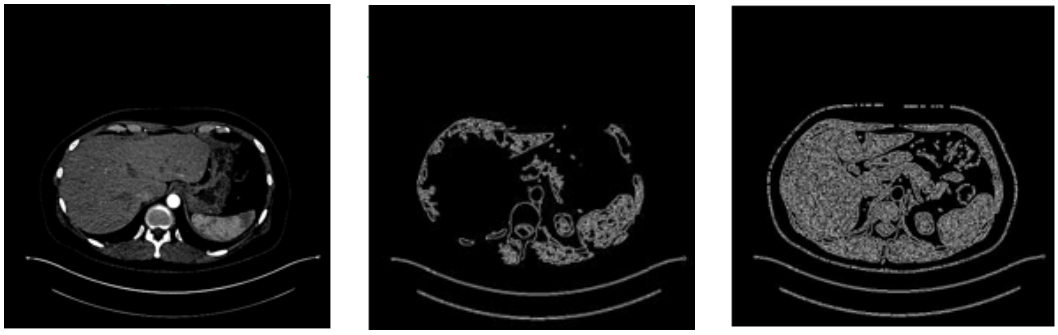
$G(x, y) = (1/(2\pi\sigma^2)) * exp(-(x^2 + y^2)/(2\sigma^2))$.

where x and y are the coordinates of the filter, and σ is the standard deviation of the Gaussian distribution.

Step 2 : Calculating the intensity gradient of the image
The convolved image is smoothed out with Sobel kernel in both horizontal and vertical direction to get first derivative in horizontal direction (Gx) and vertical direction (Gy).These are used to find gradient magnitude and direction
$magnitude = sqrt(Gx^2 + Gy^2)$
$Direction = atan2(Gy, Gx)$.

Step 3:Non-Maximal Suppression
The gradient magnitude is thinned to a one-pixel wide line by keeping only the local maxima in the gradient direction. This is done by comparing the gradient magnitude at a pixel with its two neighboring pixels in the direction of the gradient.

Step 4:Hysteresis Thresholding
The two values; MinVal and MaxVal are used and the following conditions are applied.
if Intensity of a point is greater than maxval then confirmed edge
if Intensity of a point less than minval then discard
if Intensity of a point less maxval but greater than minval check for connectivity with sure edge else discarded

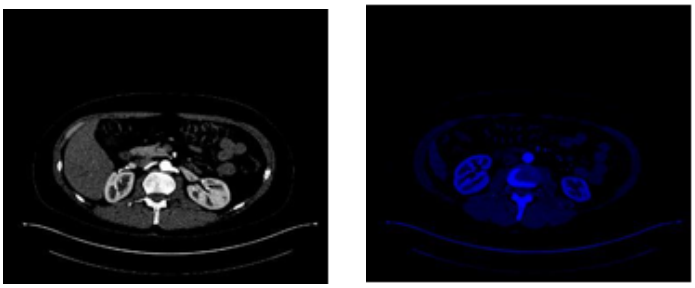


(a) (b) (c)

Figure 6: Pre-processing using Canny Edge Detection (a) Input image (b) Edge detection including kidney area c) Edge detection including all features.

Scenario 2: Pre-processing using Hue, Saturation, Value (HSV) Hue, Saturation, and value are three different hyper-parameters that make up a color space used in image processing to segment an object based on variation in color, texture, and grey intensity. Hue refers to the color and is measured in degrees. It represents the wavelength of the light in the visible spectrum in an image. Saturation measures the purity or greyness of a grayscale image. It is measured as percentage purity with 0 percent being least saturated and 100 percent being fully saturated. The value defines the brightness or intensity of a color in an image. A value of 0 percent refers to a dark image (black) whereas 100 percent refers to the full bright color in the image. The features were captured using the OpenCV function.

scenario 3: Pre-processing using Affine Transformation The shape, size, and orientation of objects in an image can be changed using the fundamental approach known as affine transformation. It enables us to change an object while maintaining its straight lines and planes. The objective is to map points in the source coordinate system to the locations in



(a) (b)

Figure 7: Pre-processing using HSV (a) Input image (b) Output Image

the destination coordinate system where those points correlate. An essential part of defining the transformation is the affine transformation matrix, which is frequently represented as a 2x3.matrix. There are six values in it, and they define how the change is carried out. Scaling factors, rotation angles, translational distances, and shearing factors are represented by these quantities.

- We can resize an object either uniformly or unevenly by scaling it. In contrast to non-uniform scaling, which can cause an object to be stretched or compressed in different directions, uniform scaling increases an object's size while keeping its shape.
- An object is rotated by turning it around a predetermined point in the image. We can adjust the object's orientation.
- When an object is translated, it is moved in a certain direction without changing its size or shape. It enables us to reposition an object within the image.
- An alteration that tilts an object's shape is called shearing. It modifies the angles at which the object's lines are spaced apart.

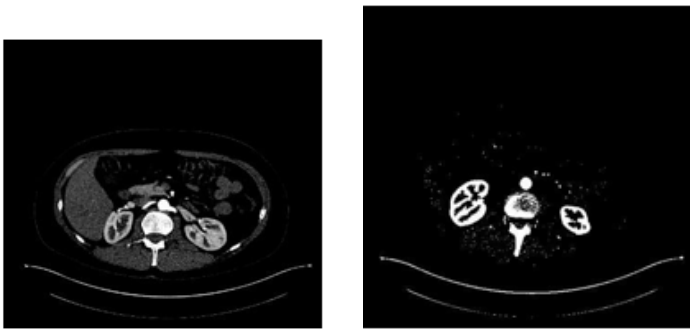
We do transformations such as simultaneous scaling, rotating, translating, and shearing by combining these operations using the affine transformation matrix. This enables us to adjust objects in photos, fix distortions, align pictures, and make other geometric changes. The benefit of affine transformations is that they are computationally effective and may be used to transform complete images or big groups of points.

Scenario 4: Pre-Processing using Binary Threshold Segmentation is a common pre-processing technique used in image analysis. This technique uses a threshold value of T to partially segment an image. Using a gray-scale image and a fixed threshold, all pixels above the threshold are set to 1 and all pixels below the threshold are set to 0. In this way, a binary image on pixel values 1 and 0 is created. For our experiment, a threshold value of 100 using hit and trail to primarily focus on the lower part of the image which generally includes segmentation output.

5. ANALYSIS FOR EXPERIMENTS

5.1. Model training using different hyper parameters

From the series of experiments above, it can be seen that the increased depth of UNET gave better results in terms of the number of epochs taken to train the model. One reason for decreased epochs is that the model has a smaller filter in the very first encoder which captured features closer to pixel level and the image becomes more simplified by the time it reaches the bottleneck of the model. The results were slightly better for UNET with dropouts in terms of



(a) (b)

Figure 8: Pre-processing using Binary Threshold Segmentation (a) Input image (b) Output Image

training epochs as the dropout layers set the non-significant inputs to a layer to 0 randomly with the frequency of rate. The significant inputs are scaled up by a factor of 1/ (1 - rate). This prevents overfitting of the model and stops the training earlier. By increasing the depth of the model, the model now has an additional layer of encoders and decoders. Hence, for this reason, the number of filters in each layer was reduced. This means that in each layer only the selected features will be captured. Decreasing the number of filters for UNET and dropout UNET also affected the results. The number of trainable and non-trainable parameters of the three models are shown below:

Table 7: Classification with VGG 16, segmenting with UNET

Model	**Trainable Parameters**	**Non-Trainable Parameters**
U-NET	3,835,553	2,944
Dropout U-NET	3,835,553	2,944
Increased Depth U-Net	962,985	1,504

For the experiment using pre-trained model, the total computation cost is high as the training of the model includes the cost of first classifying the segmented and non-segmented images and then adds the cost of segmentation by UNET.

5.2. Model Training with Pre-Processing Techniques

In this iteration, 7 different experiments were carried out. U-NET and increased depth U-NET were both trained on non-normalized datasets without feature selection to observe their execution time, dice coefficient, number of epochs after which learning rate drops, and loss curve. All experiments were carried out with epochs parameter set to 15, learning rate set to 0.1, and GPU P100.

Comparison of loss The labels such as affine and canny represent the pre-processing technique applied on increased depth U-NET. The loss for U-NET and increased depth U-NET is relatively similar at all epochs with initial values being 0.30 and 0.32 respectively. The lines for both epochs have completely overlapped after 11 epochs. The loss for each epoch in feature upscaling techniques is also similar even though the loss is greater than no feature selection technique applied. For affine transformation, binary threshold, and canny with all features, the decrease in loss results in the partially overlapping curve, the hsv curve overlaps after 3 epochs

but the canny with limited curve never overlaps. The initial loss for Canny with edge detection with limited and all features, HSV, threshold, and affine is 0.70,0.68,0.76,0.69,0.66 respectively

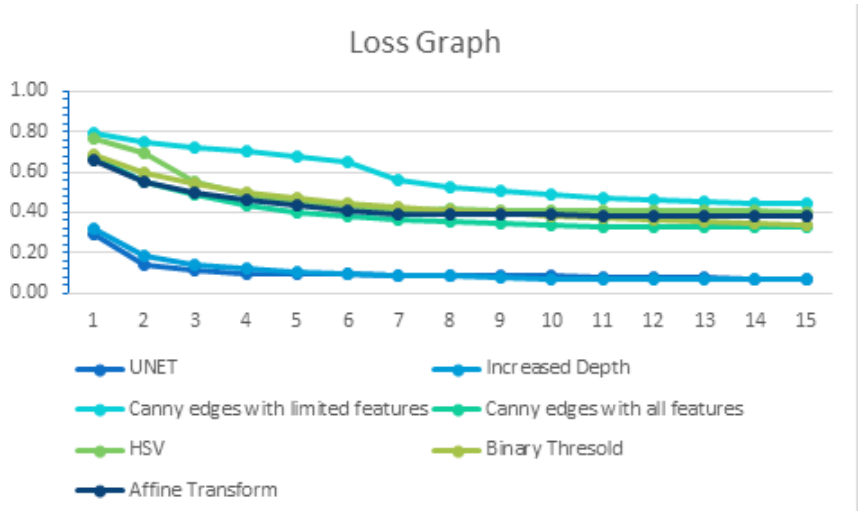


Figure 9: Comparison of loss of all experiments

Comparison of Dice score A similar pattern was observed in the dice coefficient as with the loss curve. The dice score obtained after 15 epochs ranges from 0.55 for canny edge detection with limited features to 0.93 for U-NET and increased depth U-NET model. The performance of improving the dice coefficient is comparatively similar for U-NET and increased Depth U-NET. Similar patterns in dice scores improvement of Canny with all edges, HSV, Binary threshold, and Affine transformation can also be seen in the graph. The U-NET model achieved a 0.93 score after 11 epochs whereas the increased depth model achieved a similar dice score after 9 epochs. The dice score for Canny with all edges, HSV, Binary threshold, and Affine transformation are relatively similar with values 0.68,0.60,0.66,0.62 respectively.

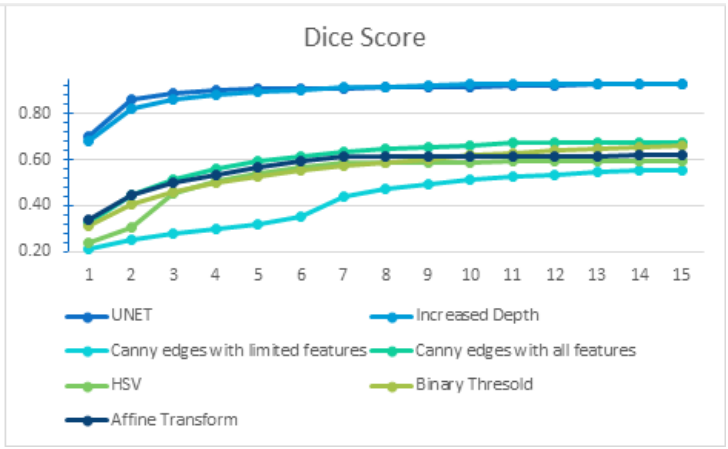


Figure 10: Comparison of dice score of all experiments

Comparison of Execution time U-NET model took maximum time to complete execution followed by an increased depth U-NET model. Canny Edge with limited features outperformed all other models and preprocessing techniques by taking a minimum time of 3958s to complete

the execution. Edges provide a simplified representation of an image and highlight only the significant features while ignoring the rest resulting in lower dimensionality which contributes to lower execution time. The hsv technique outperforms at being the second best after canny edge detection with limited features as viewing images in HSV color space simplifies complex images. On the other hand, The U-NET and increased depth of both models trained on higher dimensionality images resulting in maximum execution time.

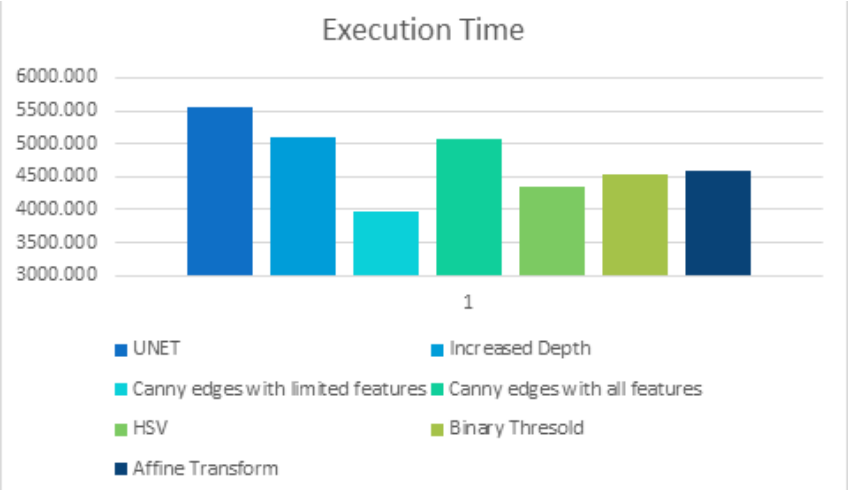


Figure 11: Comparison of the execution time of all experiments

Comparison of Epochs The number of epochs here is counted till a point where the learning rate dropped to a minimum. As shown in the above graph, the model trained with HSV pre-processing took the minimum number of epochs whereas the binary threshold technique took maximum epochs.

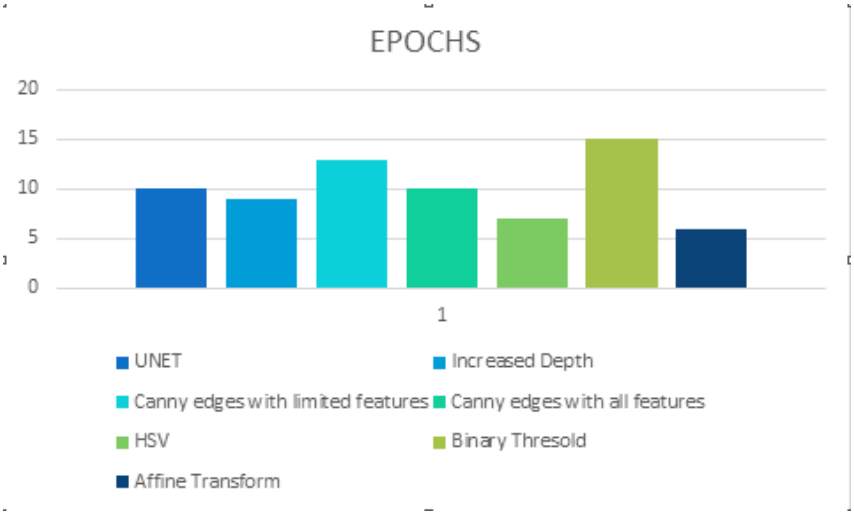


Figure 12: Comparison of epochs taken of all experiments

Discussion and Limitations Different preprocessing feature selection techniques were tried and tested. All feature upscaling techniques resulted in similar results in terms of dice score and

loss but have undoubtedly resulted in lesser execution time as compared to the state-of-the-art U-NET model. The reason for the higher loss may be the fact that the number of segmented images is very less than the number of non-segmented images so the dice score decreases when predicting each segmented image. Also, the selection of minimum features has also resulted in feature loss which also contributes to the higher loss.

CONCLUSION In conclusion, the domain of medical image segmentation is an important computer vision task with a variety of uses, including robotic surgeries, disease detection, and diagnosis. Due to its capacity to capture both local and global properties of images, the U-NET model has become a well-known deep-learning architecture for image segmentation. The encoder-decoder structure of the U- NET model, which includes skipping connections, enables it to mix high-resolution feature maps from the encoder with low-resolution feature maps from the decoder, improving the model's capacity to recognize minute objects and features in images. However, the increased computation cost of the model adds to its drawbacks. This paper applies different empirical experiments and techniques to reduce the computation cost of the U-NET model and applies different preprocessing techniques for early disease detection and prevention of chronic kidney disease. The work could be expanded for decreasing the computational cost of the model for different organs of the body and also for further reduce the loss achieved by models used in this paper.

Acknowledgments
The authors would like to thank the management of NUCES-FAST and the University Malaya for supporting the resources used for experimental execution.

Authors' Contributions
Uzair Iqbal conceptualized and supervised the research, ensuring a well-structured workflow. Bisma carried out the experimental work focused on hyperparameter tuning of the UNET architecture and contributed to the technical documentation of the experiments. Javaria contributed to the training and evaluation of the UNET with classifiers, as well as the technical documentation related to these experiments. Manahil conducted experiments involving UNET with feature engineering and contributed to the technical writing of the experimental processes and results.

Clinical Trial Number
Not applicable.

Funding
There is no funding support for this project.

Competing Interests
The authors declare no competing interests.

Availability of Data and Material
The data and materials are available at: `https://drive.google.com/drive/folders/1o1BXWG0M_YIC7-bwtdtY5EDqL6ZpNLQ2?usp=sharing_eil_m&ts=6343a1c7`

Ethical Approval Statement
This research work was reviewed by the ethical board of biomedical research at FAST-NUCES, Pakistan. Additionally, this work satisfies the parameters of the Declaration of Helsinki.

Consent for Publication

I, the undersigned, give my consent for the publication of identifiable details, which can include photograph(s), videos, case history, and/or details within the text ("Material") to be published in the above journal and article.

References

[1] Chronic Kidney Disease in the United States, 2021. Atlanta, GA: US Department of Health and Human Services, Centers for Disease Control and Prevention; 2021.

[2] Romagnani, Paola; Remuzzi, Giuseppe; Glassock, Richard; Levin, Adeera; Jager, Kitty J.; Tonelli, Marcello; Massy, Ziad; Wanner, Christoph; Anders, Hans-Joachim (2017). Chronic kidney disease. Nature Reviews Disease Primers, 3(), 17088–. doi:10.1038/nrdp.2017.88

[3] Webster, Angela C; Nagler, Evi V; Morton, Rachael L; Masson, Philip (2016). Chronic Kidney Disease. The Lancet, (), S0140673616320645–. doi:10.1016/S0140-6736(16)32064-5

[4] https://www.kidney.org/atoz/content/gfr

[5] Dan Li 1, 2. C. (2022, 12). Deep Segmentation Networks for Segmenting Kidneys and Detecting Kidney Stones in Unenhanced Abdominal CT Images. doi: https://doi.org/

[6] Wenshuai Zhaoa, D. J. (2020). MSS U-Net: 3D segmentation of kidneys and tumors from CT images with a multi-scale supervised U-Net. 19. doi:https://doi.org/10.1016/j.imu.2020.100357

[7] Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., . . . Xu, D. (2022). UNETR:Transformers for 3D Medical Image Segmentation.

[8] Liu, Ying, Hui Cui, Tiangang Zhang, Toshiya Nakaguchi, and Ping Xuan. 2021. "Integrating Channel Context Attention and Regional Association Attention for Kidney and Tumor Segmentation."

[9] Chen, Zhang, Zhiqiang Tian, and Jihua Zhu. 2022. "C-CAM: Causal CAM for Weakly Supervised Semantic Segmentation on Medical Image."

[10] Shi Yin , Qinmu Peng , Hongming Li , Zhengqiang Zhang , Xinge You , Katherine Fischer , Susan L. Furth , Gregory E. Tasian , Yong Fan.(2019)." Automatic kidney segmentation in ultrasound images using subsequent boundary distance regression and pixelwise classification networks."

[11] Ooi, Alexander Ze Hwan, et al. "Interactive Blood Vessel Segmentation from Retinal Fundus Image Based on Canny Edge Detector." 2021, https://www.mdpi.com/1424-8220/21/19/6380.

[12] Singh, Pritpal. "A neutrosophic-entropy based clustering algorithm (NEBCA) with HSV color system: A special application in segmentation of Parkinson's disease (PD) MR images." 2020, https://www.sciencedirect.com/science/article/abs/pii/S016926071931822X.

[13] https://doi.org/10.48550/arXiv.1505.04597

[14] The TensorFlow Authors

[15] opencv.org

[16] Wolfram Research (2008), GaussianFilter, Wolfram Language function, https://reference.wolfram.com/language/ref/GaussianFilter.html (updated 2016).

[17] Heller, Nicholas and Isensee, Fabian and Maier-Hein, Klaus H and Hou, Xiaoshuai and Xie, Chunmei and Li, Fengyi and Nan, Yang and Mu, Guangrui and Lin, Zhiyong and Han, Miofei and others (2020) The state of the art in kidney and kidney tumor segmentation in contrast-enhanced CT imaging: Results of the KiTS19 Challenge , Medical Image Analysis , Elsevier.

[18] Heller, Nicholas and Sathianathen, Niranjan and Kalapara, Arveen and Walczak, Edward and Moore, Keenan and Kaluzniak, Heather and Rosenberg, Joel and Blake, Paul and Rengel, Zachary and Oestreich, Makinna and others (2019)The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes ,arXiv preprint arXiv:1904.00445