Course Project Report on

UNeXt: MLP-based Rapid Medical Image Segmentation Network

Supervisor

Prof. Suyash Awate

Submitted By

Nitish Ganagwar - 203050069

Shreyas Narahari - 203050037

Computer Science And Engineering (CSE)

Indian Institute of Technology Bombay

April 2022

Introduction:

One of the major tasks in medical imaging is segmentation, which is used in many sectors for diagnosis. UNet is one of the pathbreaking architectures that showed how efficient encoder-decoder convolutional networks with skip connections are. UNet also requires much fewer data as other segmentation networks require thousands of images to be trained well.

We chose to implement the paper “UNeXt: MLP-based Rapid Medical Image Segmentation Network” (Link), which proposes a segmentation network called UNext, a derivative of UNet like other popular networks such as UNet++, UNet3+, Etc.

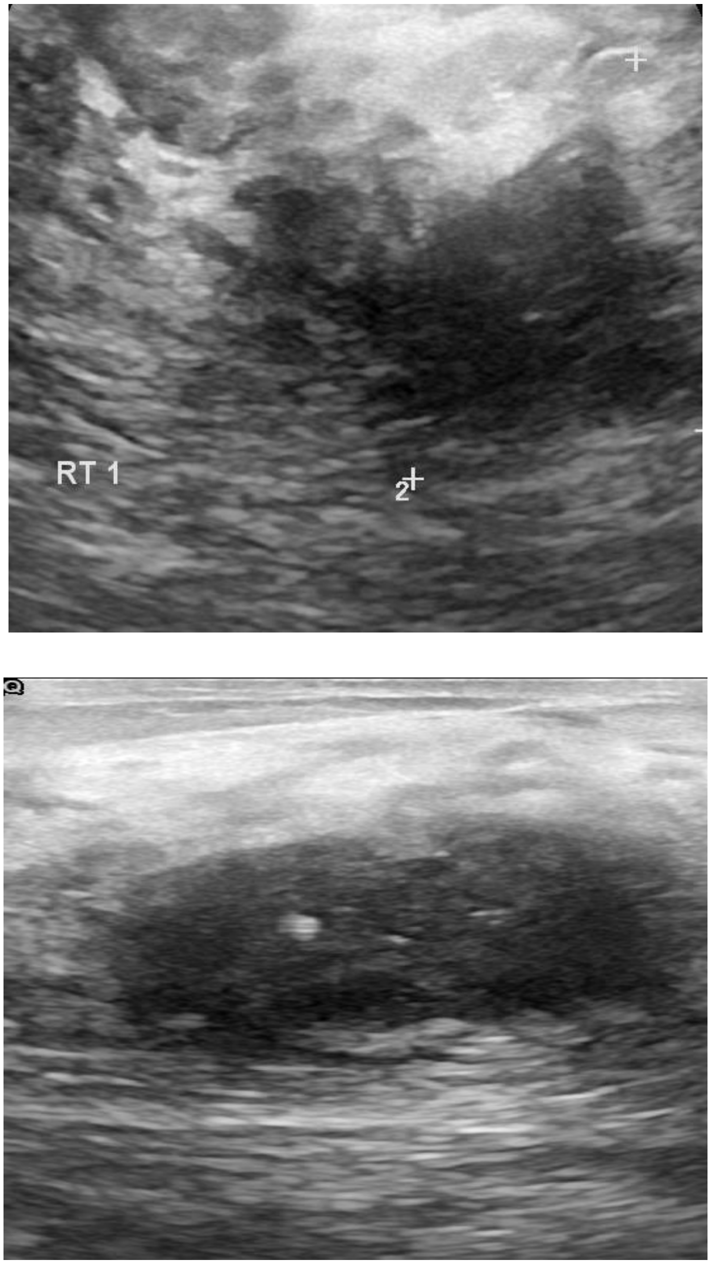
We have compared the UNext network against the original UNet network.

We have trained the UNext network given at the repository (Link) by the authors. Before training the network, we have performed the necessary preprocessing of the images to be accepted by the network.

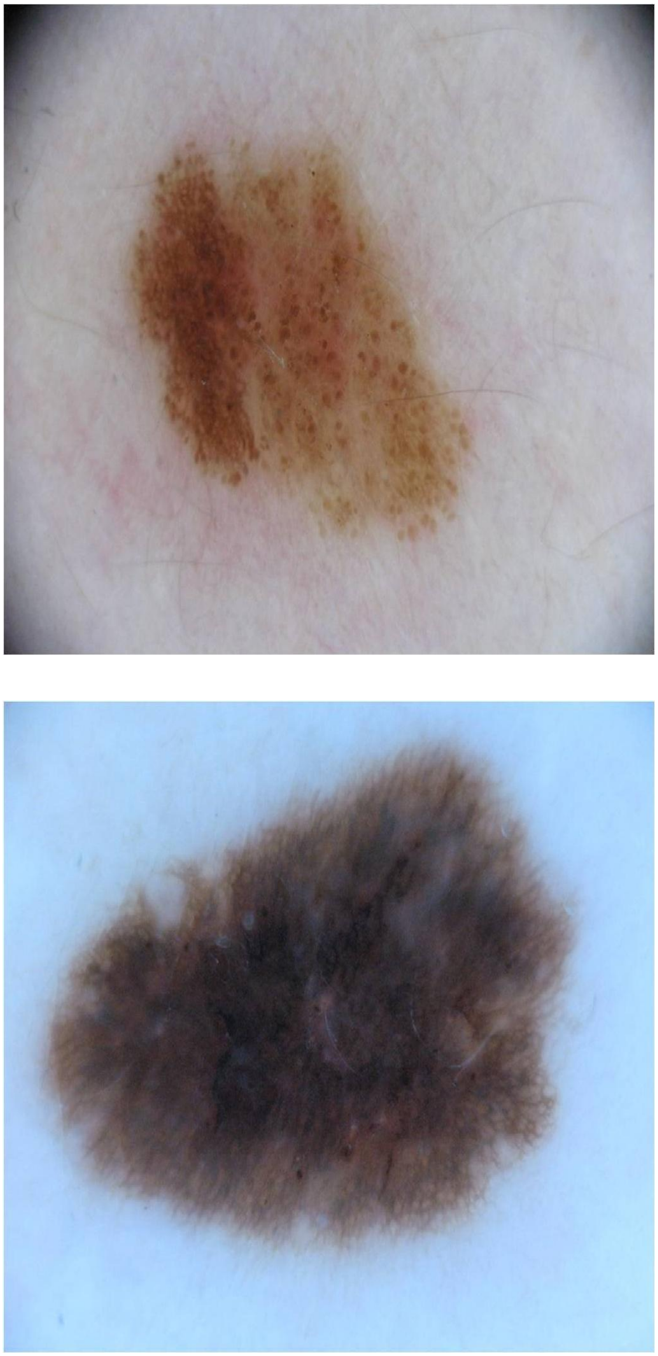
For training the UNet network, we have written our training and preprocessing code.

Dataset:

Two datasets have been used for training. The first one is the International Skin Imaging Collaboration(ISIC 2018) dataset (Link), which consists of 2594 images, and the second dataset is the Breast UltraSound Images (BUSI) dataset (Link), which consists of 647 images. The datasets were resized to 256 x 256 for both the networks, and an 80:20(Train: Validation) split was used. A batch size of 8 was used throughout the training.

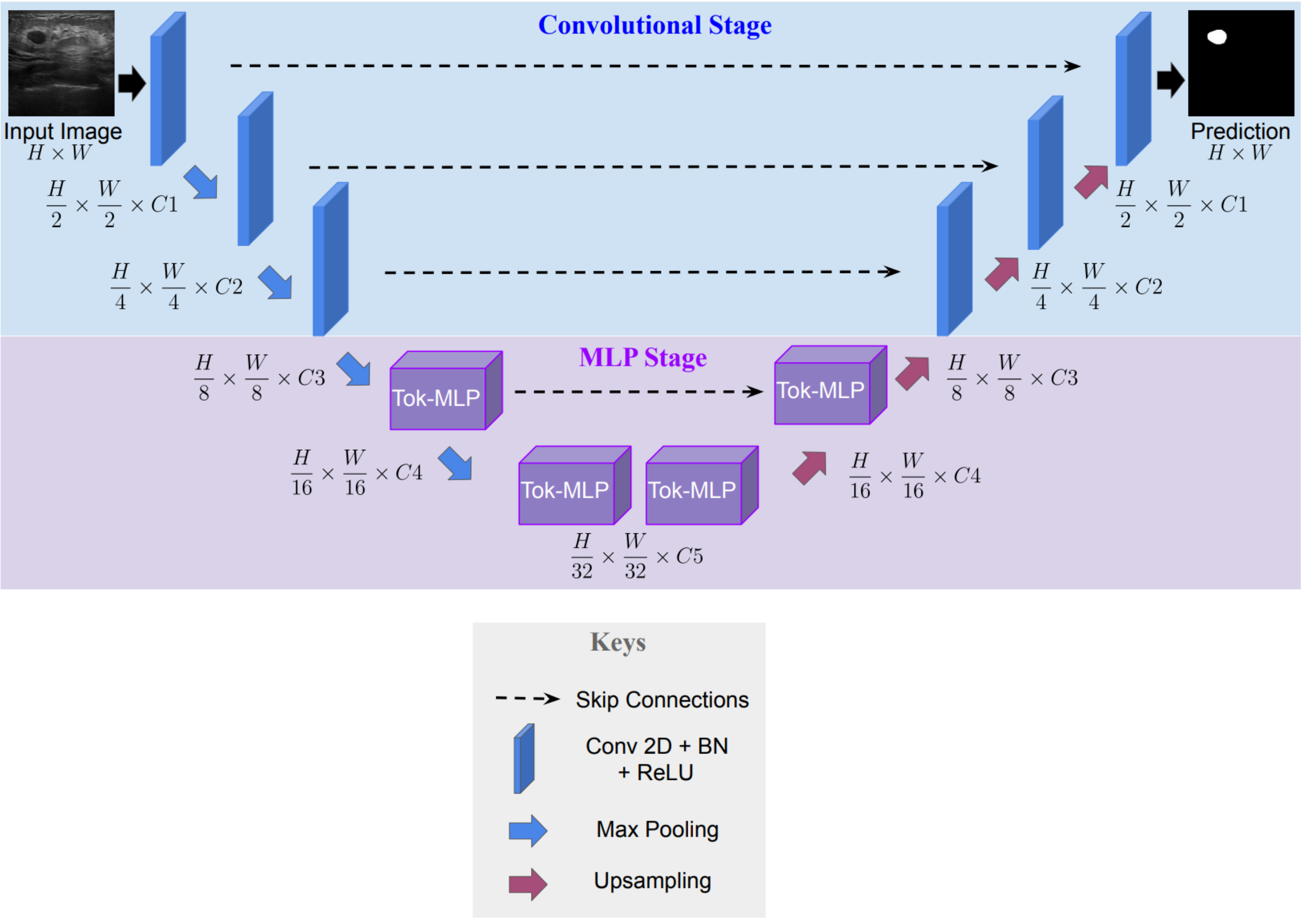


Sample images are taken from BUSI dataset



Sample images are taken from ISIC dataset

UNext Architecture:


Convolutional Stage
Input Image
$H \times W$
Prediction
$H \times W$
$\frac{H}{2} \times \frac{W}{2} \times C1$
$\frac{H}{4} \times \frac{W}{4} \times C2$
$\frac{H}{2} \times \frac{W}{2} \times C1$
$\frac{H}{4} \times \frac{W}{4} \times C2$
MLP Stage
$\frac{H}{8} \times \frac{W}{8} \times C3$
Tok-MLP
Tok-MLP
$\frac{H}{8} \times \frac{W}{8} \times C3$
$\frac{H}{16} \times \frac{W}{16} \times C4$
Tok-MLP
Tok-MLP
$\frac{H}{16} \times \frac{W}{16} \times C4$
$\frac{H}{32} \times \frac{W}{32} \times C5$
Keys
Skip Connections
Conv 2D + BN + ReLU
Max Pooling
Upsampling

The image shows the network architecture of the UNeXt model

UNext has a two-stage encoder-decoder architecture similar to UNet, with the two stages as:

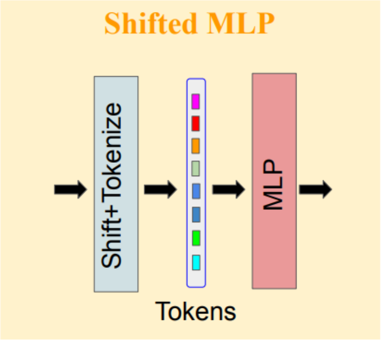
1) Convolution Stage.
2) Tokenized MLP Stage.

The image is given as input to the encoder. In the encoder, it Passes through 3 convolutional blocks, and two tokenised MLP blocks. The decoder has two tokenised MLP blocks followed by three convolutional blocks. Skip connections are also present between the encoder and decoder. Channels are denoted as C1, C2, C3, C4, C5.

Convolution Stage:

In the convolution stage, each block contains,

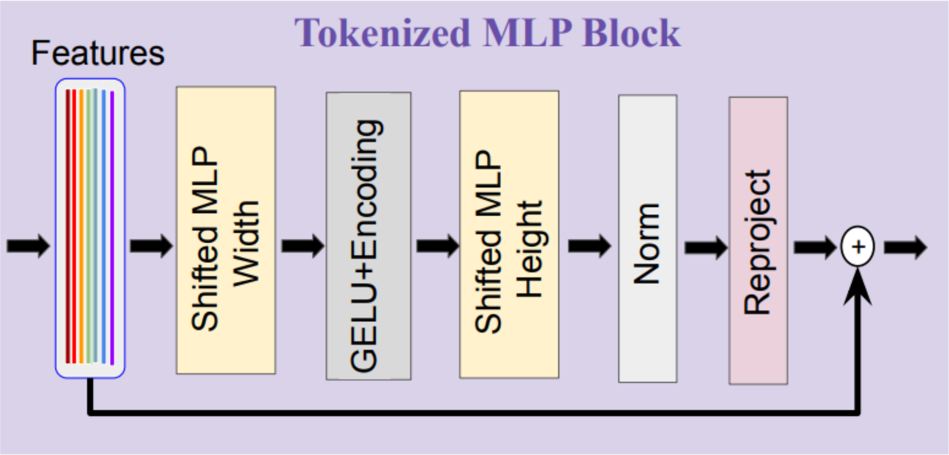
1. Convolutional layer:
    a. With kernel size of 3x3, stride and padding are at 1.
2. Batch Normalization Layer.
3. ReLU activation layer.
4. For upsampling/downsampling:
    a. Encoder: Uses Max-pooling layer with 2x2 pool window.
    b. Decoder: Uses bilinear interpolation layer for upsampling instead of transpose convolutions. The reason behind using bilinear interpolation is that it requires less number of parameters. Hence making the technique work fast.



The image denotes the steps involved in shifted MLP

Shifted MLP Stage:

In the Shifted MLP stage, shifts are done to induce locality along the axes. The features are shifted along the width axis first and then across the height axis. Where features are split into h different partitions and are shifted by j locations according to the specified axis.



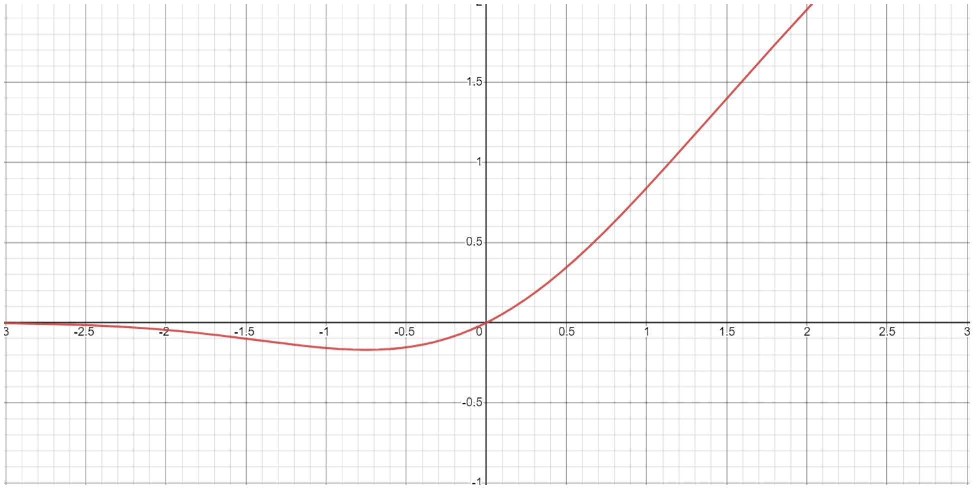
The image denotes the steps involved in Tokenized MLP block.

Tokenized MLP Stage:

In the Tokenised MLP stage, the features are shifted and projected onto tokens. The number of channels changed to embedding dimension (E), the number of tokens. These tokens are passed to a Shifted MLP layer which shifts them across the width. The output features are passed on to the depth-wise convolutions layer(DW-Conv) for two reasons:

1. It helps to encode the positional information. It performs better than the standard encoding techniques.
2. It uses fewer parameters and then also provides enhanced performance.

GELU is used here instead of ReLU as it was found to perform better. The function is represented below,



The GELU activation function is defined as

$$\mathrm{GELU}(x) := xP(X \leq x) = x\Phi(x).$$

Here Φ(x) is the standard gaussian cumulative distribution function.

$$\begin{gathered}
X_{shift} = Shift_W(X); T_W = Tokenize(X_{shift}), \\
Y = f(DWConv((MLP(T_W)))), \\
Y_{shift} = Shift_H(Y); T_H = Tokenize(Y_{shift}), \\
Y = f(LN(T + MLP(GELU(T_H)))),
\end{gathered}$$

The above image shows the mathematical implementation of the shifting MLP part of the network. Here, the shift over X has been made along the width denoted as Shift_W(X), followed by tokenization. The tokenised part is given as input to depth wise convolution for further processing. Obtained output is denoted with Y. Now the second shift is performed along with the height, the output obtained is denoted as Y_shift. Which is further tokenized and is given as input to Multi-Layer Perceptron. Where the residuals are added along with the output from the MLP, which is further layer normalised as batch normalisation at this point does not mean a lot.

Result:

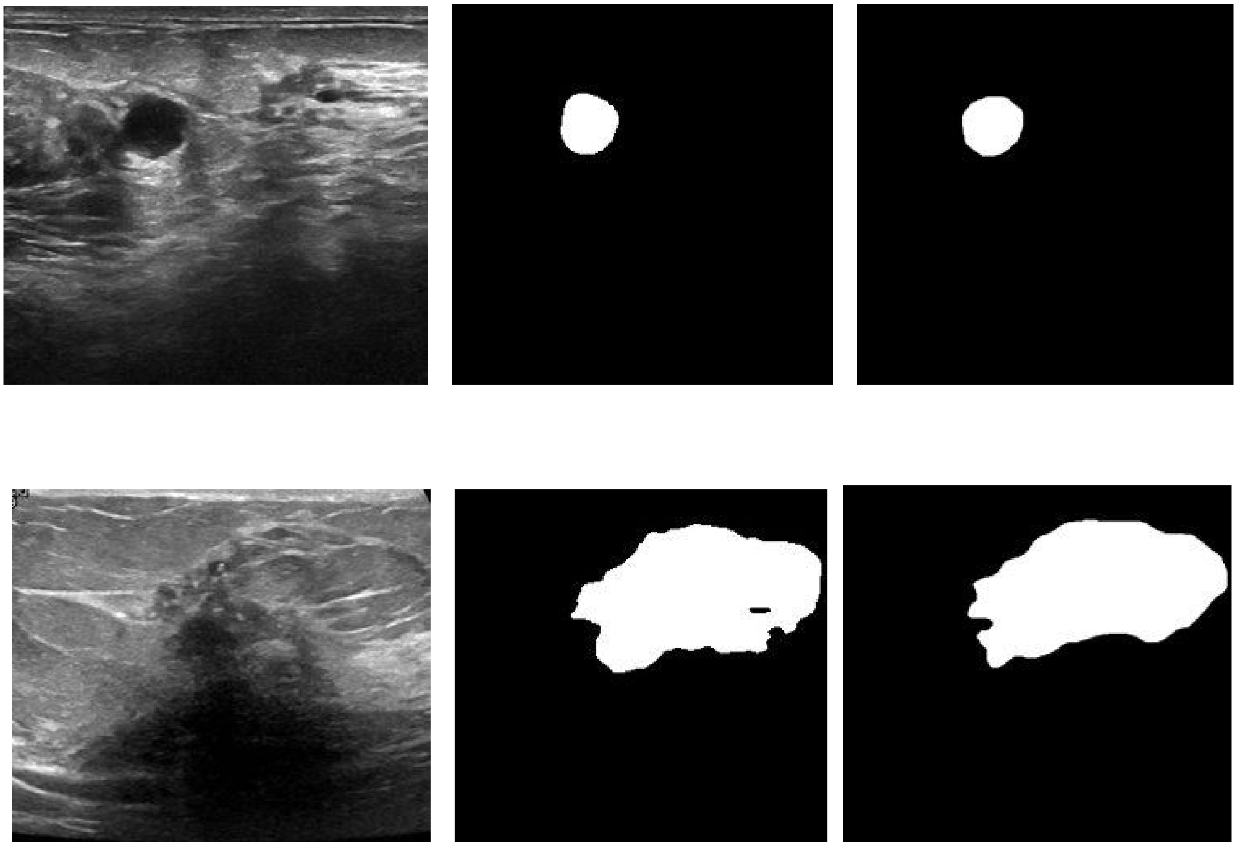
Networks	**ISIC (F1-Score)**	**ISIC (IoU)**	**BUSI (F1-Score)**	**ISIC (IoU)**
UNet	0.48	0.52	0.88	0.90
UNeXt	0.94	0.82	0.95	0.63

After training, we tested our trained model over test data. We split the original dataset into an 80:20 ratio. We used IoU (Intersection over Union) and F1 score as the metric for evaluating our trained model performance. Each of these metrics is shown above in the table for the respective dataset, i.e. ISIC and BUSI.

Segmentation masks predicted using our model over some of the test images are shown below. The images shown below are predicted using UNeXt and UNET. They are shown separately below.

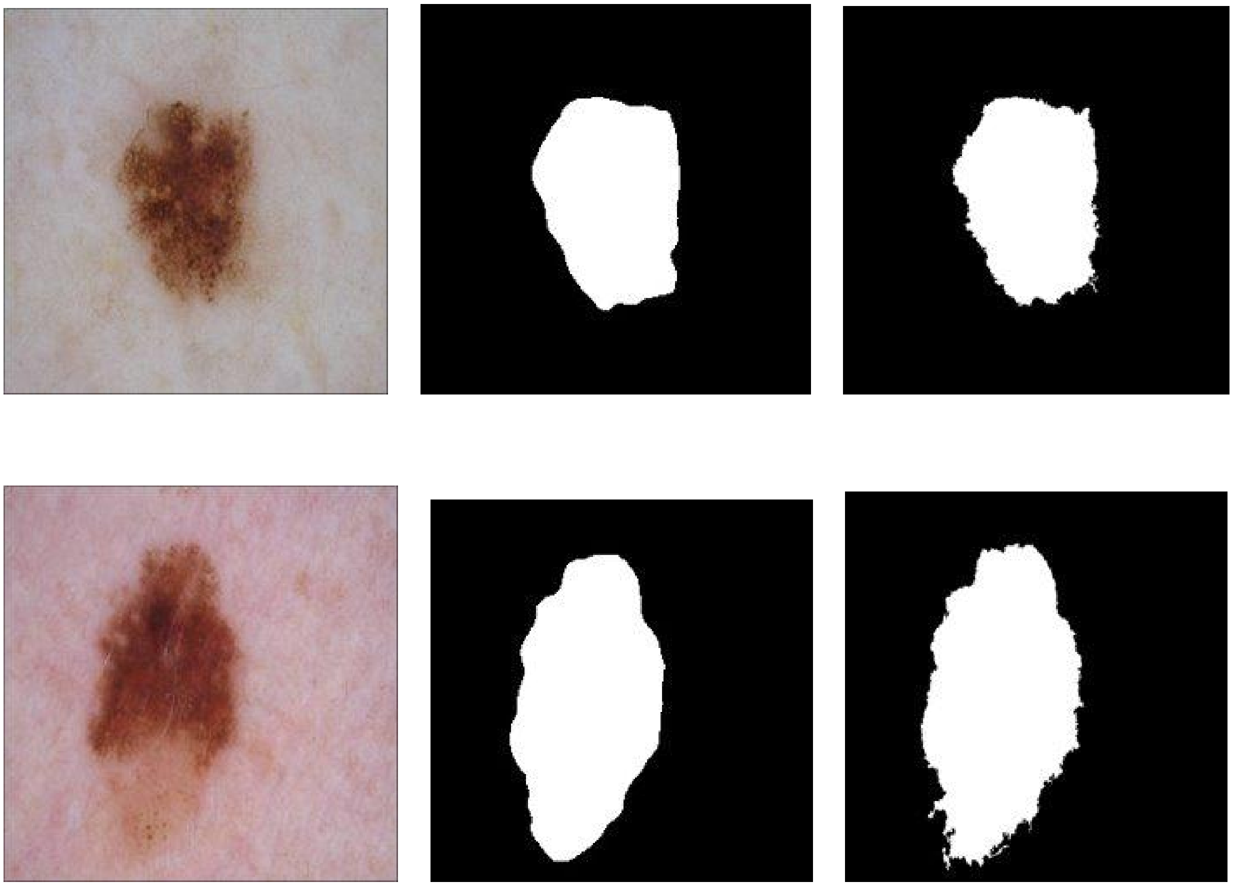
Results for UNext:

(BUSI dataset)



Test images taken from the BUSI dataset, where the first column image is the original image, the second column image is the predicted output using the trained model and the third column image is the original segmented mask of the image.

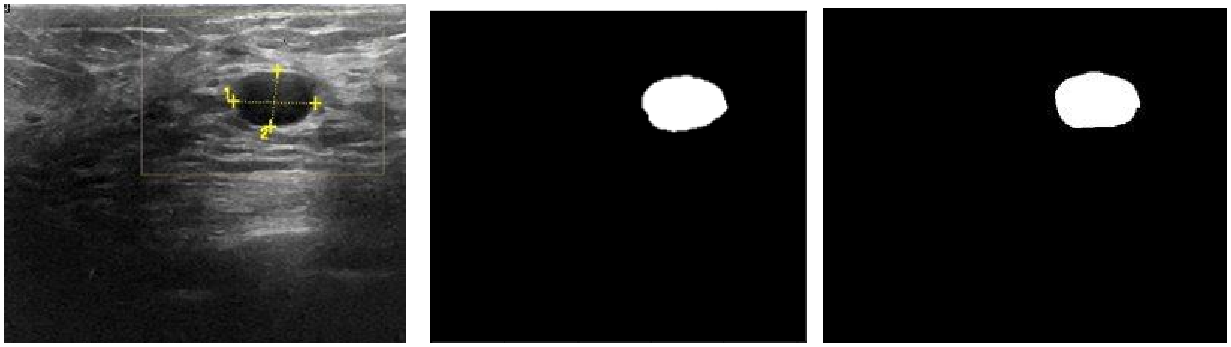
(ISIC 2018 dataset)

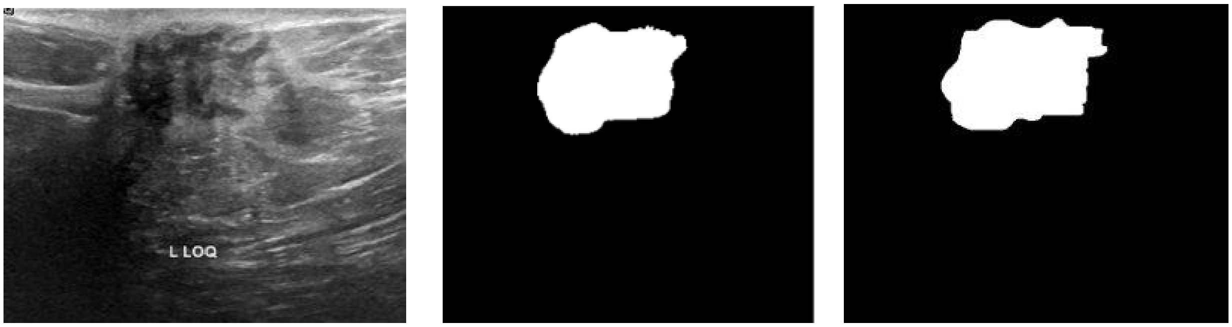


Test images taken from the ISIC dataset, where the first column image denotes the original image, the second column image denotes the predicted mask using the trained model and the third column image denotes the original segmented mask of the image.

Results for UNet:

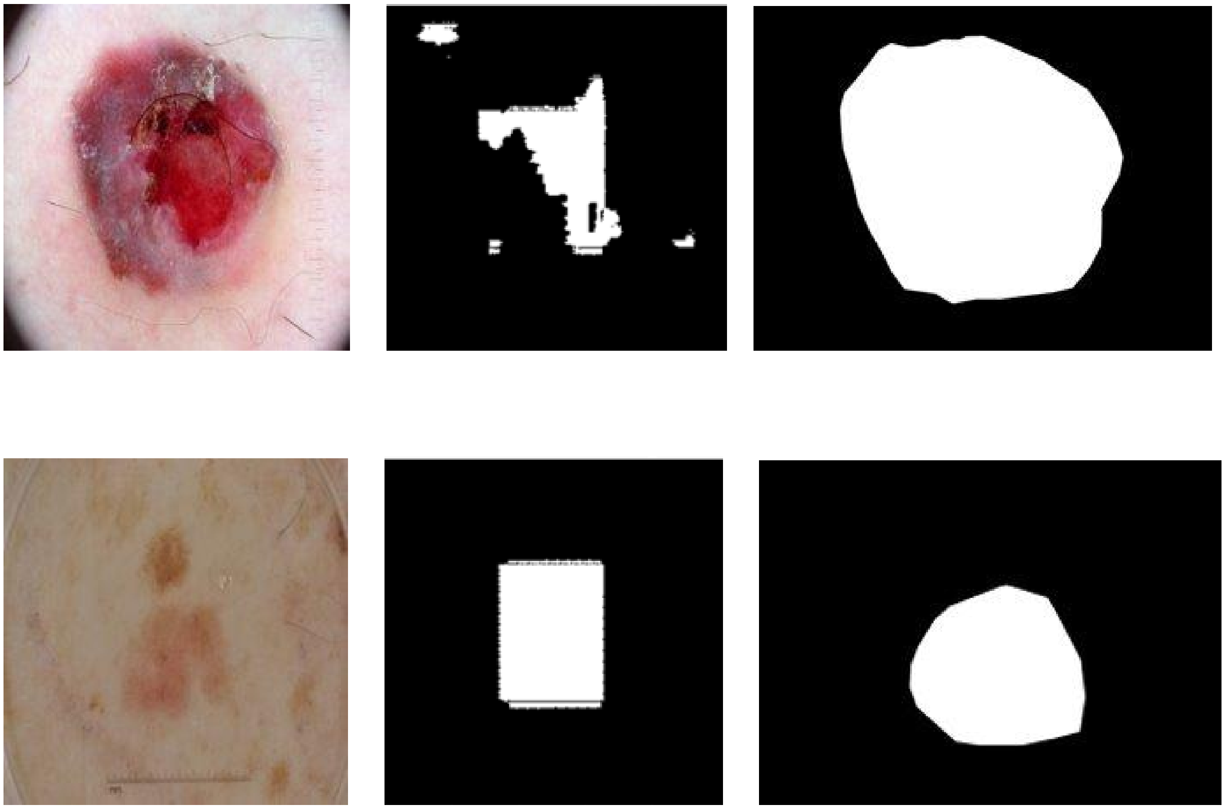
(BUSI dataset)





Test images taken from BUSI dataset, where the first column image denotes the original image, the second column image denotes the predicted mask using the trained model and the third column image denotes the original segmented mask of the image.

(ISIC 2018 dataset)



Test images taken from the ISIC dataset, where the first column image denotes the original image, the second column image denotes the predicted mask using the trained model and the third column image denotes the original segmented mask of the image.

Conclusion:

We have trained two models, UNeXt and UNET, on two datasets, ISIC and BUSI. We have used the F1 score and IoU as the metrics for testing our network's efficacy.

As in the table shown above, the UNeXt model clearly outperforms UNET, and the images generated using the UNeXt model are nicely segmented for both the datasets i.e. ISIC and BUSI. Through our experience, we have observed that output generated using UNeXt is quite fast compared to UNET.

UNET performance is quite good on the BUSI dataset but not that good in ISIC dataset which can clearly be seen from the table shown above as well as in the images shown above.

References

1. Valanarasu, Maria Jose, and Vishal M. Patel. “[2203.04967] UNeXt: MLP-based Rapid Medical Image Segmentation Network.” *arXiv*, 9 March 2022, https://arxiv.org/abs/2203.04967..
2. Ronneberger, Olaf, et al. “U-Net: Convolutional Networks for Biomedical Image Segmentation.” *Computer Vision Group, Freiburg*, May 2015, https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/.
3. Jose, Jeya Maria. “jeya-maria-jose/UNeXt-pytorch: Official Pytorch Code base for "UNeXt: MLP-based Rapid Medical Image Segmentation Network."” *GitHub*, https://github.com/jeya-maria-jose/UNeXt-pytorch.
4. “ISIC Challenge Datasets.” *ISIC Challenge*, https://challenge.isic-archive.com/data/.
5. “Breast Ultrasound Images Dataset” BUIS Dataset, Breast Ultrasound Images Dataset | Kaggle
6. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)

7. Lian, D., Yu, Z., Sun, X., Gao, S.: As-mlp: An axial shifted mlp architecture for vision. arXiv preprint arXiv:2107.08391 (2021)