CS 736 Course Project Report

Recurrent Residual Convolutional Neural Network based on U-Net (R2U-Net) for Medical Image Segmentation

Akkapaka Saikiran Rohan Shah Rushabh Kanadiya

1 Introduction

Segmentation is the process of partitioning an image into different meaningful segments. In medical imaging, these segments often correspond to different tissue classes, organs, tumors, or other biologically relevant structures. This can be used to detect blockages or enlargements of blood vessels, brain abnormalities, tumors in lungs, abscesses in abdomen, and many other ailments.

Due to the slow process and tedious nature of manual segmentation approaches, there is a significant demand for computer algorithms that can do segmentation quickly and accurately without human interaction. In the previous decade or so, deep learning techniques have received vast attention for the task of automated image segmentation.

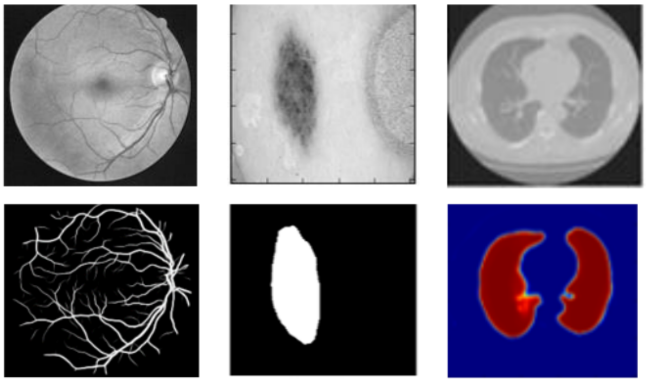


Figure 1: Segmentation of retinal blood vessels, skin cancer, and lung tumor respectively

2 Related Work

Convolutional Neural Networks

Convolutional Neural Networks are a class of deep learning algorithms most commonly applied to analysing image data. They have been around since a long time, but saw a resurgence lately and now are the state of the art algorithms for image classification, object detection and image segmentation. Their resurgence can primarily be attributed to the performance of AlexNet [1] which famously won the ImageNet challenge [2] in 2012 by a big margin. Many researchers then picked up the ideas of AlexNet and improved on it.

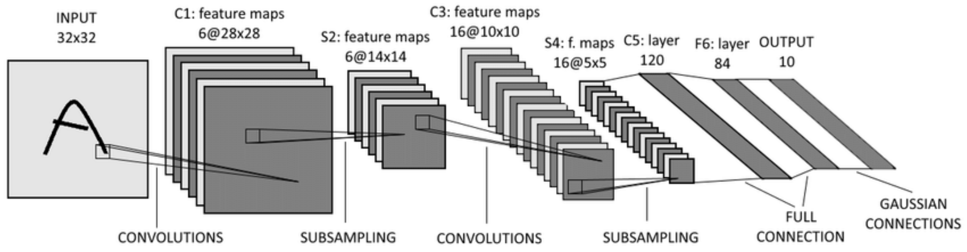


Figure 2: LeNet, 1998

From an architectural point of view, the CNN model for classification tasks requires an encoding unit and provides class probability as an output. In classification tasks, we perform convolution operations with activation functions followed by subsampling layers which reduces the dimensionality of the feature maps.

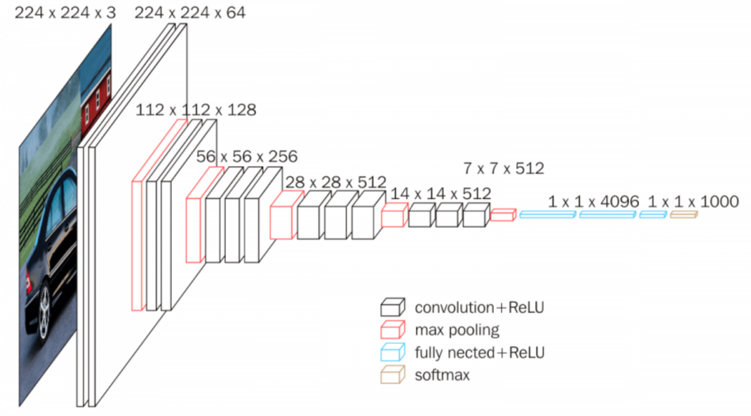


Figure 3: VGG net, 2014

Eventually, softmax operations are applied at the end of the network to compute the probability of the target classes. As opposed to classification tasks, the architecture of segmentation tasks requires both convolutional encoding and decoding units to produce segmentation maps with the same dimensionality as the original input image.

U-Net

U-Net is a convolutional neural network that was developed for biomedical image segmentation at the Computer Science Department of the University of Freiburg, Germany in 2015. The network only uses the valid part of each convolution without any fully connected layers. The main idea is to supplement a usual contracting network by successive layers, where pooling operations are replaced by upsampling (fractionally-strided or transpose convolutions) operators. U-Nets draw their name from their shape.

U-Nets have done exceedingly well since their inception and continue to be the building blocks of many state-of-the-art network architectures, outperforming most other techniques.

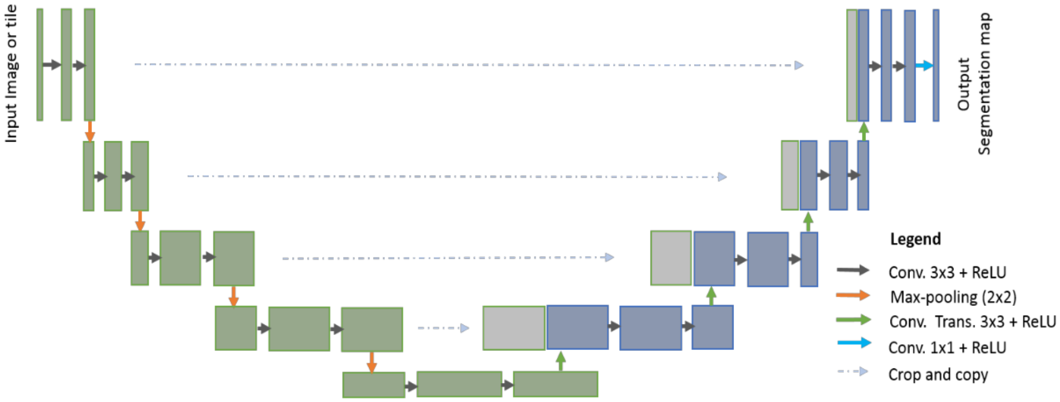


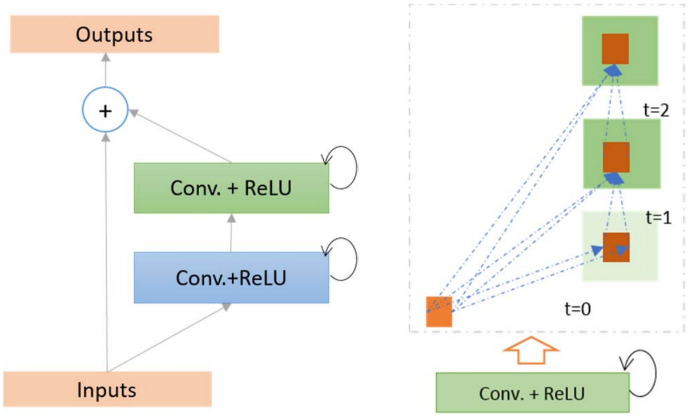
Figure 4: A standard U-Net

3 Our Architecture

We implement two modified and improved segmentation models, one using recurrent convolution networks, and another using recurrent residual convolutional networks. Our work is almost entirely based on [3].

3.1 Recurrent convolutions

The key idea of any recurrent block is to reuse weights or maps and maintain some state. In our case, we feed the output of a convolution layer back into that layer as an input a few times (parameterizing the process) before passing it on to the next layer.



(a) (b)

Figure 5: Diagrams displaying (a) the recurrent residual convolutional unit and (b) the unfolded version of the recurrent convolutional unit

3.2 Residual blocks

First introduced by Microsoft in their ResNet [4], residual connections are skip connections, which are used to allow gradients to flow through a network directly, without passing through non-linear activation functions. Non-linear activation functions, by nature of being non-linear, cause the gradients to explode or vanish (depending on the weights).

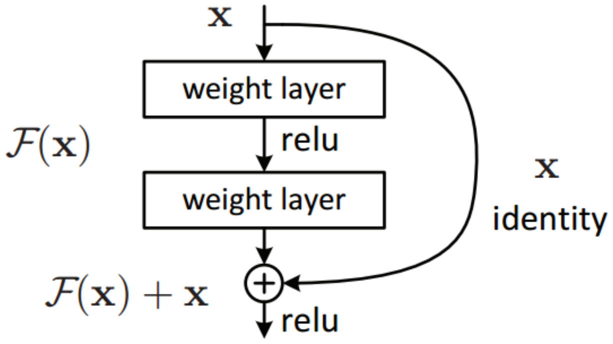

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 6: The residual block from the original paper on ResNets

If $\mathcal{F}$ is a non-linear map (corresponding to a couple of convolutions in our implementation), then the final output $\mathcal{H}$ of a residual block with input $\mathbf{x}$ would be $\mathcal{H}(\mathbf{x}) = \mathcal{I}(\mathbf{x}) + \mathcal{F}(\mathbf{x})$, where I is the Identity function, save for adjusting for shape.

3.3 Theoretical Advantages of our model

There are many benefits one can expect by employing these changes.

- A residual unit helps accelerate training and increases depth of network as opposed to width. Residual connections also alleviate the vanishing gradient problem and strengthen feature propagation through the network.
- Feature accumulation with recurrent convolutional layers ensures better feature representation for segmentation tasks, helping extract very low-level features which are essential for segmentation tasks for different modalities of medical imaging.
- The recurrent and residual operations, despite not increasing the number of network parameters, have a significant impact on training and testing performance.
- This model doesn't crop-and-copy, it uses just concatentions while making the skip connections. resulting in better feature transfer.

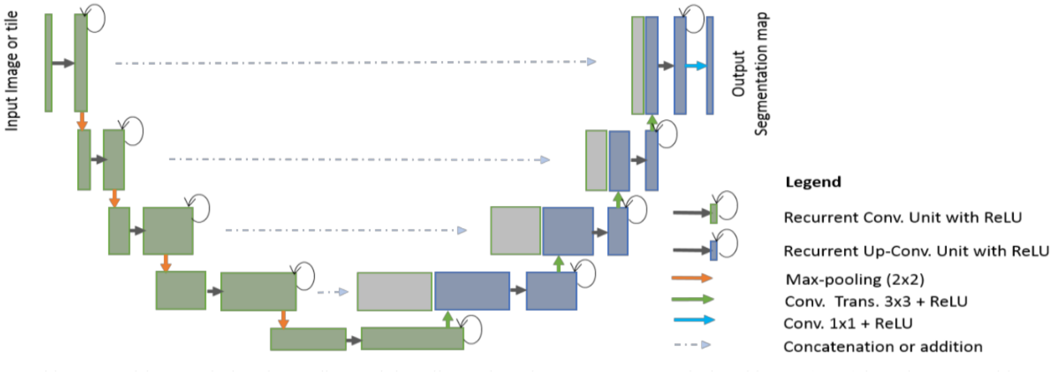


Figure 7: An RU-Net, i.e Recurrent U-Net

3.4 The nitty-gritties

- The main convolutions are all 3×3 convolutions with padding and stride 1, and the max pooling which follows it has pool size 2.
- We up-sample using 3×3 transpose convolutions with padding and stride 2, so that the activation maps double in size.
- We double the filter size of the convolutions after each max pool operation, and keep it same within a convolutional block. Based on some experimentation, we let the initial number of filters be 16, and the depth of the nets be 4. So the number of channels of the activation maps follow $1 \rightarrow 16 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 16 \rightarrow 2$.

- We set $t = 2$ for the recurrent block, meaning the output of of a Convoulutional layer is passed back into it twice.
- To add the input to the output of a (recurrent) convolutional unit, we apply 1×1 convolutions to handle the disparity in the number of feature maps.
- We use ReLU activation throughout for non-linearity after each convolution.
- After every convolution we apply batch normalisation before the non-linear activation function. Dropout is performed in the network after each block for regularization to attempt to mitigate over-fitting.

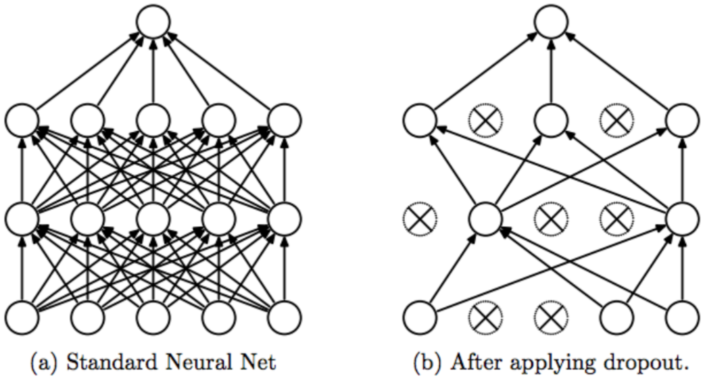


Figure 8: Dropout, visualized

The implementation is in python and uses the keras module of tensorflow. The optimization used for compilation is Adam, with a learning rate of 1e-3. The loss function chosen is Cross Entropy, and we also employed the use of class weights to tackle the class imbalance problem.

4 Dataset details

4.1 Skin Cancer Segmentation

This dataset is taken from the Kaggle competition on skin lesion segmentation that occurred in 2017. This dataset contains 2000 training samples, 150 validation samples, and 600 testing samples. The original size of each sample was 700×900, which was re-scaled to 256×256 for this implementation. Normalisation is not performed on these images as batch normalisation takes care of them.

4.2 Retinal Blood Vessels Segmentation

We pooled three different popular datasets for retina blood vessel segmentation for this task. These were DRIVE [5], STARE, [6] and CHASE, in total amounting to 68 annotated images. These images varied in their sizes so we resized them all to

256×256. Inspired by [3], we used a patch-based method for segmenting these images and generated 11500 128×128 patches. We did this because these images have finer features compared to the skin cancer images.

We did not use any image from the above set for testing, for that we used 20 unannotated images from . Hence we only report training and validation accuracies for this dataset, along with figures showing segmentation of some images from the test sample.

5 Evaluation Metrics

We employed several metrics for evaluation. Accuracy as a metric is not suitable for semantic segmentation because many pixels of the original image may belong to the same class. In particular for our 2-class segmentation, most of the pixels in both datasets belonged to the background class, so the accuracy would be high if the networks developed a bias towards the background class. In fact, we were achieving accuracy around 80% for a network which predicted all pixels as belonging to the background class.

Therefore we implemented metrics which make sense for this task, and we list those metrics, along with their mathematical formulation below. Let TP, TN, FP, and FN denote true positive, true negative, false positive, and true negative respectively. Further, let GT refer to the ground truth image (annotation matrix) and let SR refer to the segmentation result matrix.

- Accuracy = $\frac{TP+TN}{TP+TN+FN+FP}$
- Specificity= $\frac{TN}{TN+FP}$
- Sensitivity or Recall = $\frac{TP}{TP+FN}$
- Precision[1] = $\frac{TP}{TP+FP}$
- F1-score = $2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision + Recall}}$
- Jaccard Similarity = $\frac{|GT \cap SR|}{|GT \cup SR|} = \frac{TP}{TP+FN+FP}$
- Dice Coefficient = $\frac{2|GT \cap SR|}{|GT|+|SR|} = \frac{2TP}{2TP+FN+FP}$

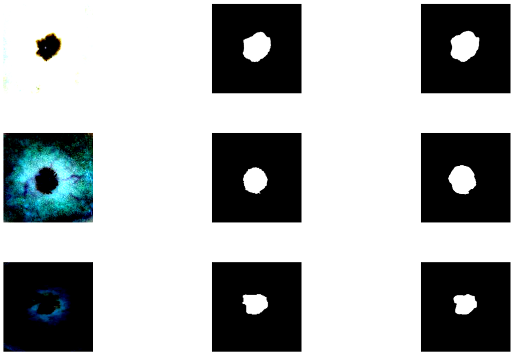
6 Experimental Results

6.1 Skin Cancer Segmention

On the Skin Cancer dataset, we trained an R2U-Net for 80 epochs with batch size set as 20. Our results are shown in table 1.

[1] We don't report this metric but it is used to calculate F1-score

Table 1: Skin Cancer Segmentation



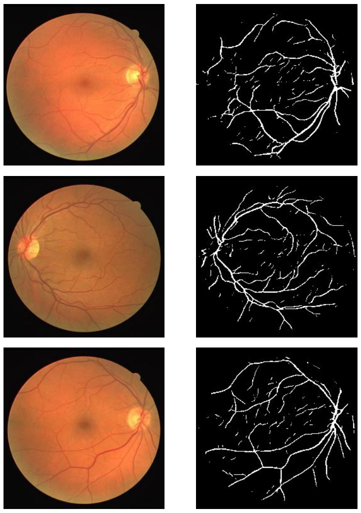
Results on a few validation images. The first column shows input images, and the second column shows ground truth, third column shows results by R2U-Net

Metric	Train	Validation
Accuracy	0.9503	0.0.8930
F1-score	0.8485	0.6348
Jaccard Similarity	0.7737	0.5609
Dice Coefficient	0.8485	0.6349
Sensitivity	0.5782	0.5782
Specificity	0.9956	0.9860

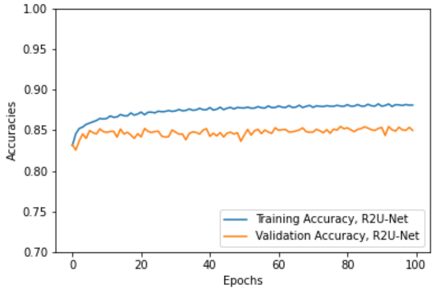
Peak values of several metrics

6.2 Retinal Blood Vessels Segmentation

Table 2: Retinal Blood Vessels Segmentation - 1



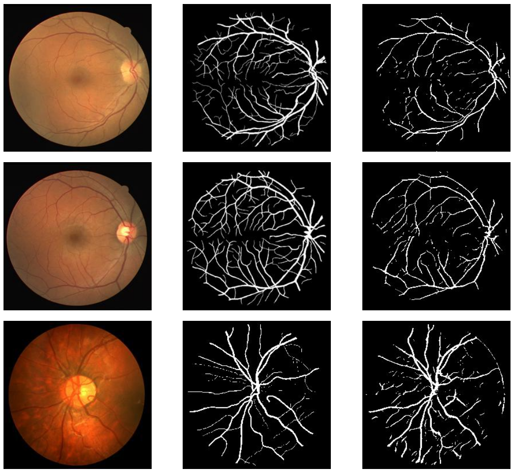
Results on a few test images. The first column shows input images, and the second column shows results by R2U-Net



Accuracies for R2U-Net

On the Retinal Blood Vessels dataset we trained an R2U-Net for 100 epochs with batch size set as 30. We used 200 steps per epoch. We show our results in tables 2 and 3.

Table 3: Retinal Blood Vessels Segmentation - 2



Results on a few validation images. The first column shows input images, and the second column shows ground truth, third column shows results by R2U-Net

Metric	Train	Validation
Accuracy	0.882333	0.854549
F1-score	0.631616	0.489198
Jaccard Similarity	0.541661	0.385061
Dice Coefficient	0.631805	0.489393
Sensitivity	0.578190	0.485667
Specificity	0.989355	0.990516

Peak values of several metrics

7 Conclusion

- R2UNET performs better than UNET and ResUNET with almost the same number of parameters.
- Data Augmentation was performed on ISIC dataset and was found to be imperative to the good results.
- Weighted Cross Entropy Loss helps tackle class imbalance problem and speeds up training.
- Accuracy was discovered to be a misleading metric because of the class imbalance problem, other metrics(Jaccard Similarity, Dice Coefficient) are better indicators of performance.
- For Implementation, pure tensorflow functions were found to be much faster because of better integration and using parallel computing, caching etc.
- Image wise standardization enhances the image contrast, and it helped us in retinal blood vessel segmentation.
- We gained an insight into SOTA Medical Image Segmentation techniques using Neural Networks.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. Curran Associates Inc., 2012.

[2] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2014.

[3] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, Tarek M. Taha, and Vijayan K. Asari. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation, 2018.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[5] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 2004.

[6] I Kanjanasurat, B Purahong, C Pintavirooj, and C Benjangkaprasert. Vascular extraction by using matched filter on retinal image. *Journal of Physics: Conference Series*, 1457:012013, jan 2020.