Prathyusha Akundi
2018701014

SegAN: Adversarial Network with Multi-scale L_1 Loss for Medical Image Segmentation

Yuan Xue, Tao Xu, Han Zhang, L. Rodney Long, Xiaolei Huang

(https://arxiv.org/pdf/1706.01805.pdf)

Prathyusha Akundi
2018701014

Abstract:

SegAN is a novel end-to-end adversarial network for the task of medical image segmentation. Since image segmentation requires dense pixel-level labelling, the single real/fake output of a classic GAN's discriminator is inefficient to produce stable and sufficient gradient feedback to the networks. In SegAN, a fully connected convolutional neural network is used as a Segmentor to generate segmentation label maps and a novel adversarial critic network with a multi-scale L_1 loss function which forces critic and segmentor to learn both global and local features that captures long- and short-range spatial relationships between the pixels is proposed. The segmentor and critic networks are trained like a min-max game: the critic is maximizing the multi-scale loss function while the segmentor is trained with the gradients passed along by the critic, with the aim to minimize the loss function. The methodology is test on ISIC skin lesion segmentation dataset.

Motivation:

Most of the existing deep learning networks designed for segmentation use pixel-wise loss function such as softmax in the last layer of their networks which lacks the ability to learn multi-scale spatial constraints directly from an end-to-end training process. Methods that use patch-based training, can avoid class imbalance problem by sampling a balanced number of patches from each class. Others, that directly employ whole images or large patches, generally resort to weighted loss function to train their networks. However, a general loss function that can enforce spatial constraints besides dealing with class imbalance problem is most desirable.

Proposed Multi-scale L1 loss function:

Given a dataset with N training images x_n and corresponding ground-truth label maps y_n, the multi-scale objective loss function L is defined as:

$$\min_{\theta_S} \max_{\theta_C} \mathcal{L}(\theta_S, \theta_C) = \frac{1}{N} \sum_{n=1}^{N} \ell_{\text{mae}}(f_C(x_n \circ S(x_n)), f_C(x_n \circ y_n)),$$

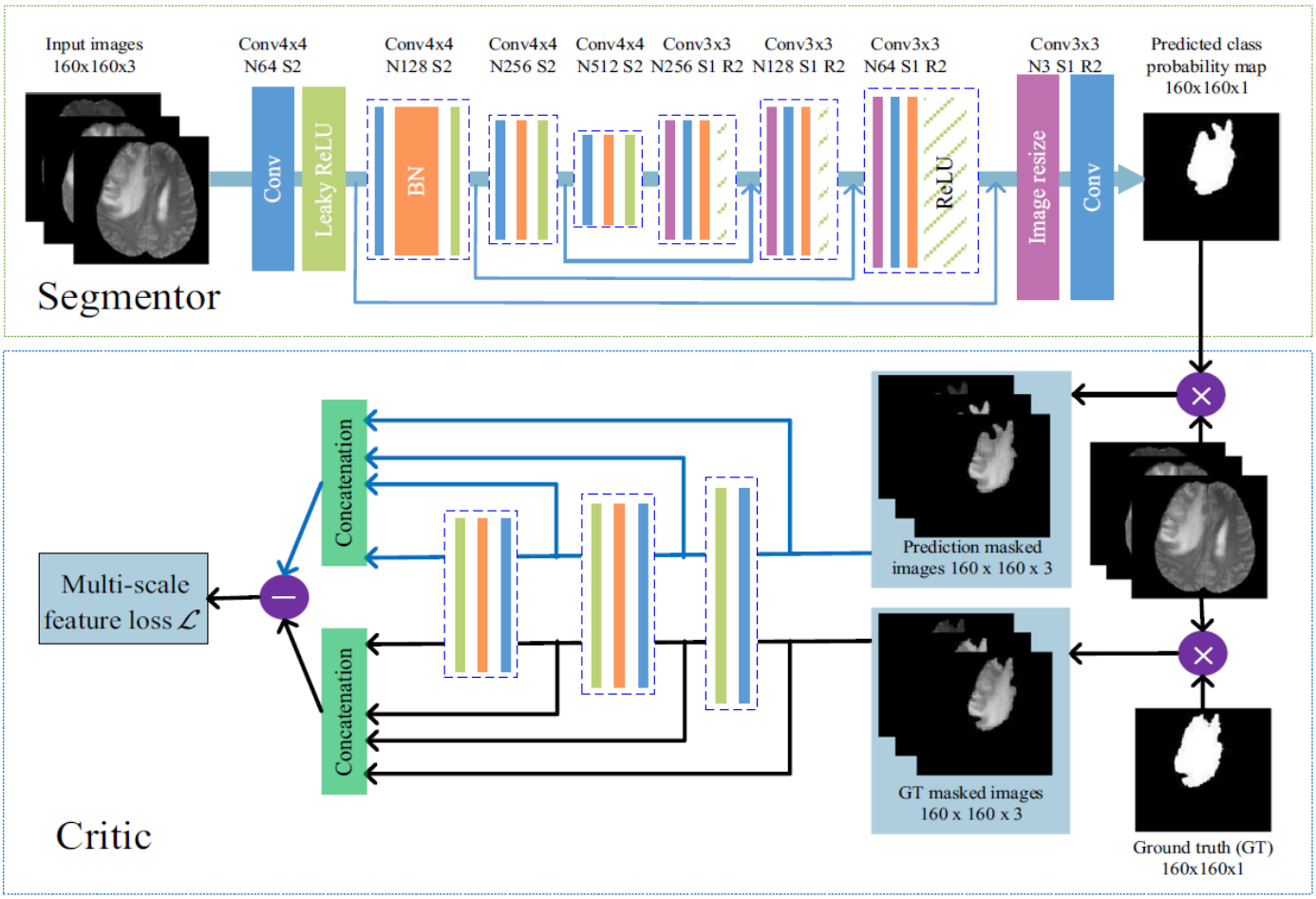
where, l_{mae} is the Mean Absolute Error (MAE) or L_1 distance; x_n o $S(x_n)$ is the input image masked by the segmentor predicted label map i.e. pixel-wise multiplication of predicted label map with the input image; x_n o y_n is the input image masked by the ground-truth. $f_C(x)$ represent the hierarchial features extracted from image x by the critic network. More specifically, the l_{mae} function can be written as:

$$\ell_{\text{mae}}(f_C(x), f_C(x')) = \frac{1}{L} \sum_{i=1}^{L} ||f_C^i(x) - f_C^i(x')||_1,$$

where, L is the total number of layers/scales in the critic network, $f_c^i(x)$ is the extracted feature map of the image x at the i^{th} layer of C.

Prathyusha Akundi
2018701014

Network Architecture:



Segmentor: A fully convolutional encoder-decoder structure is used for the segmentor ***S*** in the network. Each convolutional layer has the kernel size 4 x 4 and stride 2 for downsampling, and upsampling is performed by the image resize layer by the factor of 2 and kernel size 3 x 3 with stride 1. Skip connections are added between corresponding layers in the encoder and decoder.

Critic: The critic ***C*** has the similar structure as encoder in S. Hierarchical features are extracted from individual layers of C to compute multi-scale L_1 loss.

Dataset and other configurations:

The methodology is tested by considering skin lesion segmentation task from ISIC 2017 Part 1 challenge. The dataset consists of 2000 RGB skin images and their corresponding ground-truth masks. Out of these 2000 images, 500 are randomly used for testing. SegAN is implemented in Pytorch 1.0.1.

Pre-processing training data:

We randomly resize each training image to the size 128x 128 using bilinear interpolation method. The label-maps corresponding to each of these training images are also resized to 128 x 128 using nearest neighbour interpolation method (since we don't want to add new intensity values in the label map, we prefer nearest neighbour). For the images that are not resized, we crop them to 128 x 128 size patches. Also, horizontal and vertical flips are performed on the training images randomly.

Prathyusha Akundi
2018701014

Training:

The segmentor S and the critic C are trained by backpropagation method using the proposed multi-scale L_1 loss function. In an alternating fashion, we first fix S and train C for one step using the gradients computed from the loss function, and then, we fix C and train S for one step using the gradients computed from the same loss function passed to S from C. Here, while S tries to minimize the loss, C tries to maximize it!

As the training progresses, both S and C become more and more powerful and eventually, segmentor S will be able to produce label-maps that are very close to the ground truth as labelled by human experts.

For both Segmentor and Critic, Adam optimization method is used with beta1 = 0.5 and beta2 = 0.999. Learning rate is initially 0.002 and gets reduced for every 25 epochs. With batch-size set to 30, the number of epochs is 5000. (Note: The original paper has #epochs = 10,000. But we know that the training and validation loss are not strictly decreasing functions, we may end up having high loss by the termination of network training. Also, since the problem we are attempting i.e. skin lesion, is pretty straight forward and easy task for a deep neural network, the #epochs can be decreased.) The total training time including pre-processing is approximately 1.5 days on ADA server with 4 GPUs and 5 CPUs.

Metrics:

The following metrics are used throughout the experiment:

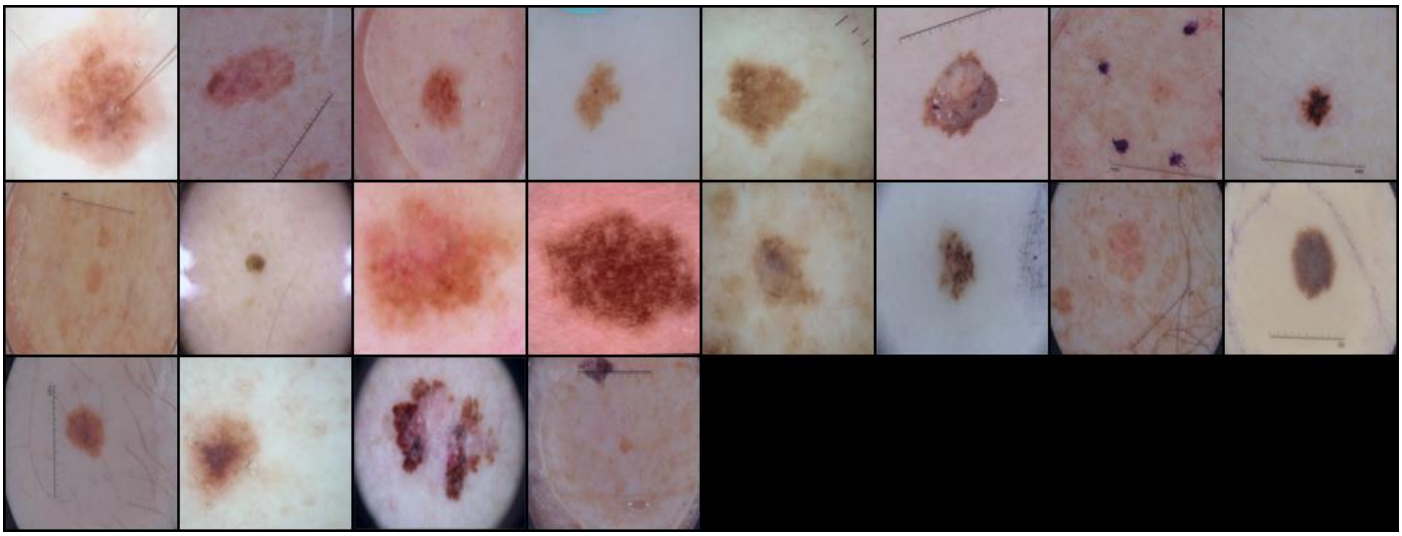
1. IOU (Intersection Over Union): Computed as IOU = $(P \cap Q)/(P \cup Q)$
2. Dice coefficient: Computed as, $2*|P \cap Q| / (|P| + |Q|)$

Results:

Segmentor S creates a label map for the image (skin lesion). The following are the results at various iterations:

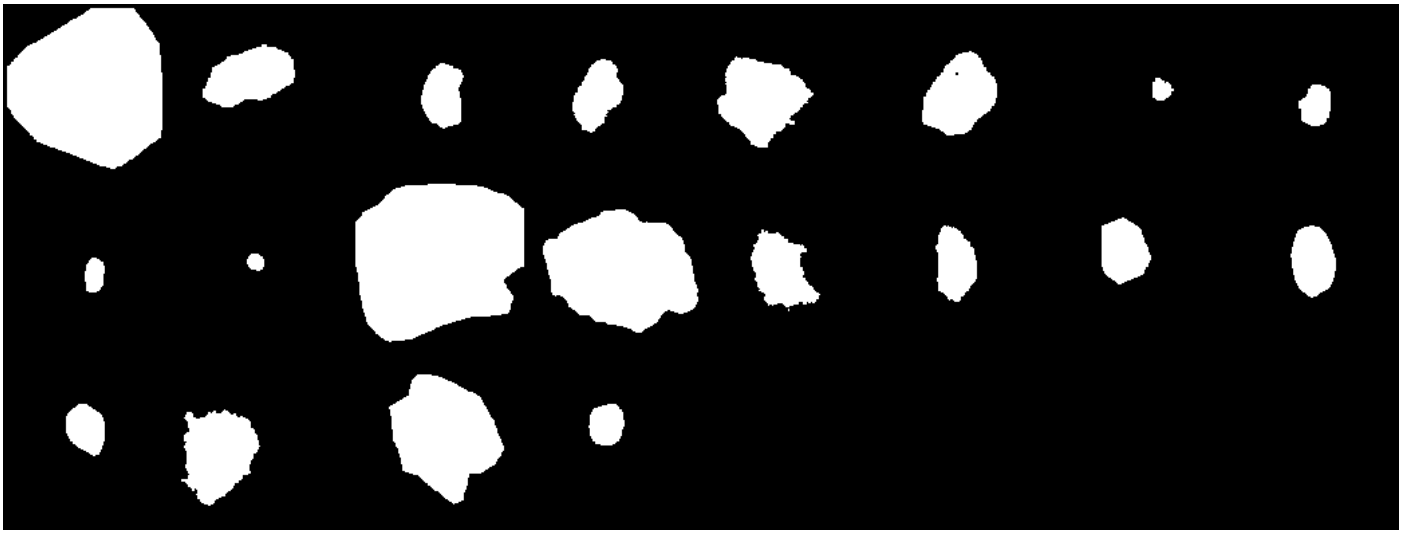
Iteration 1:

Input:

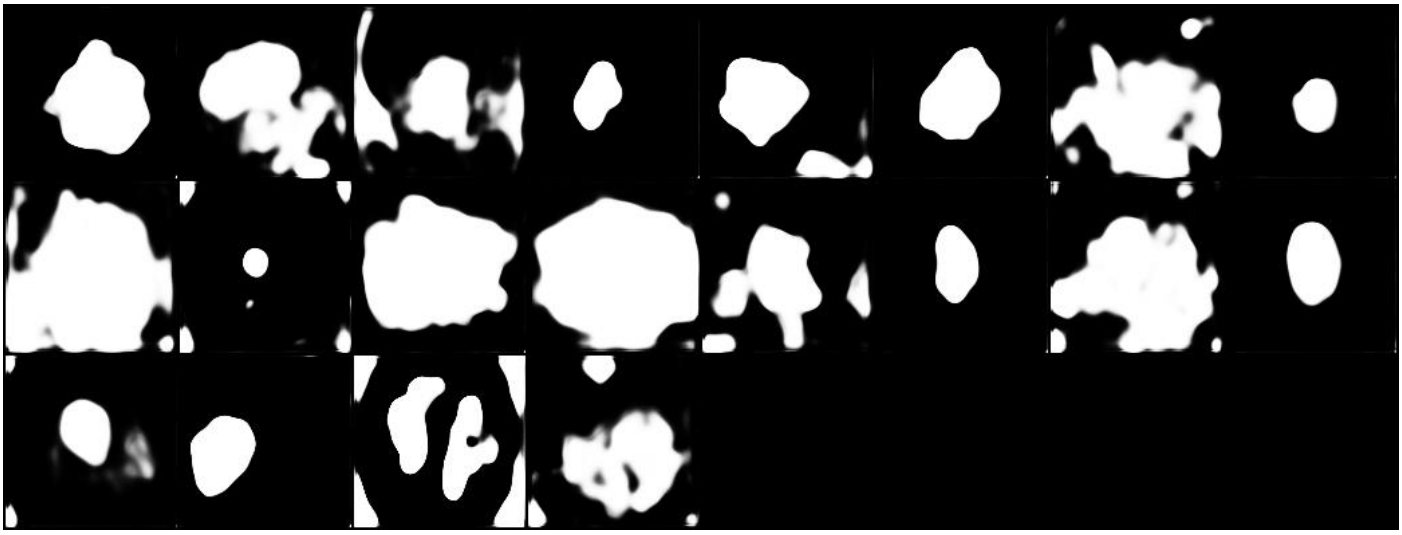


Prathyusha Akundi
2018701014

Ground truth:



Predicted label-map by Segmentor S:

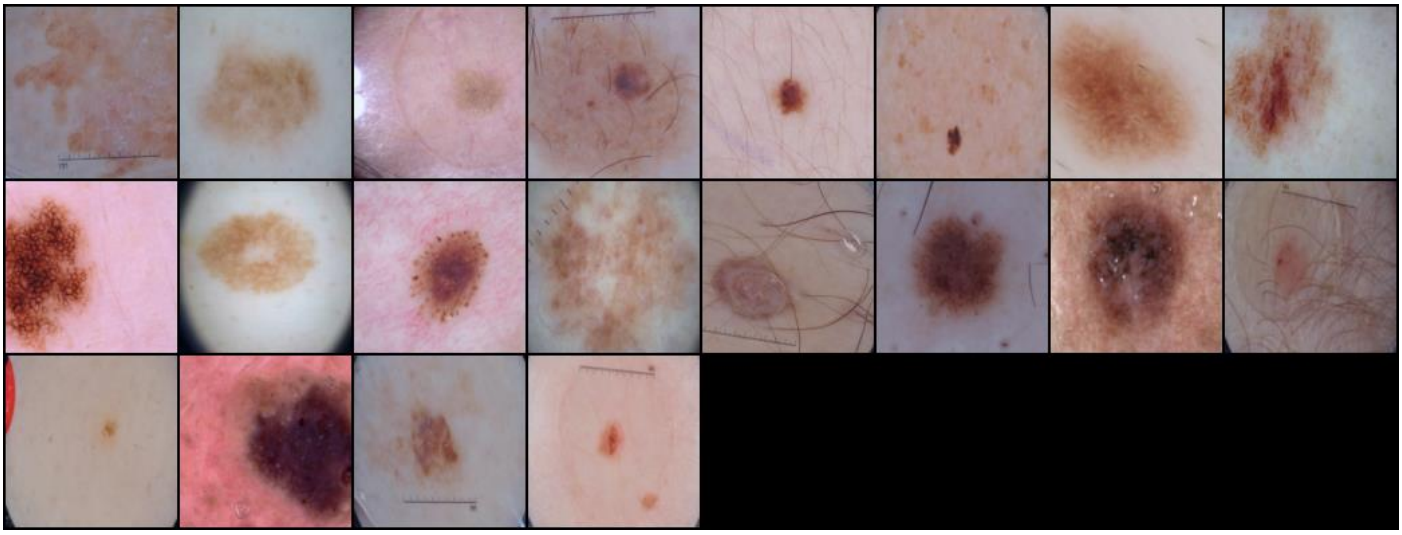


At this point, the dice coefficient is reported as: **0. 5982** and IOU as: **0.5991**

Prathyusha Akundi
2018701014

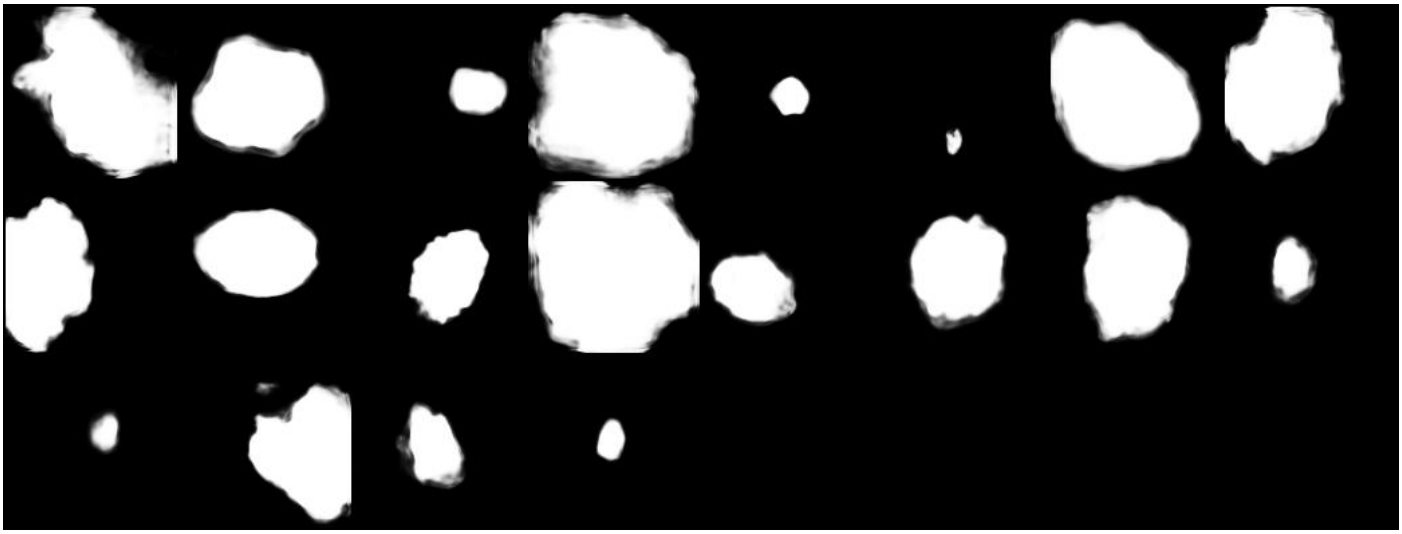
Iteration 100:

Input:



Ground truth:

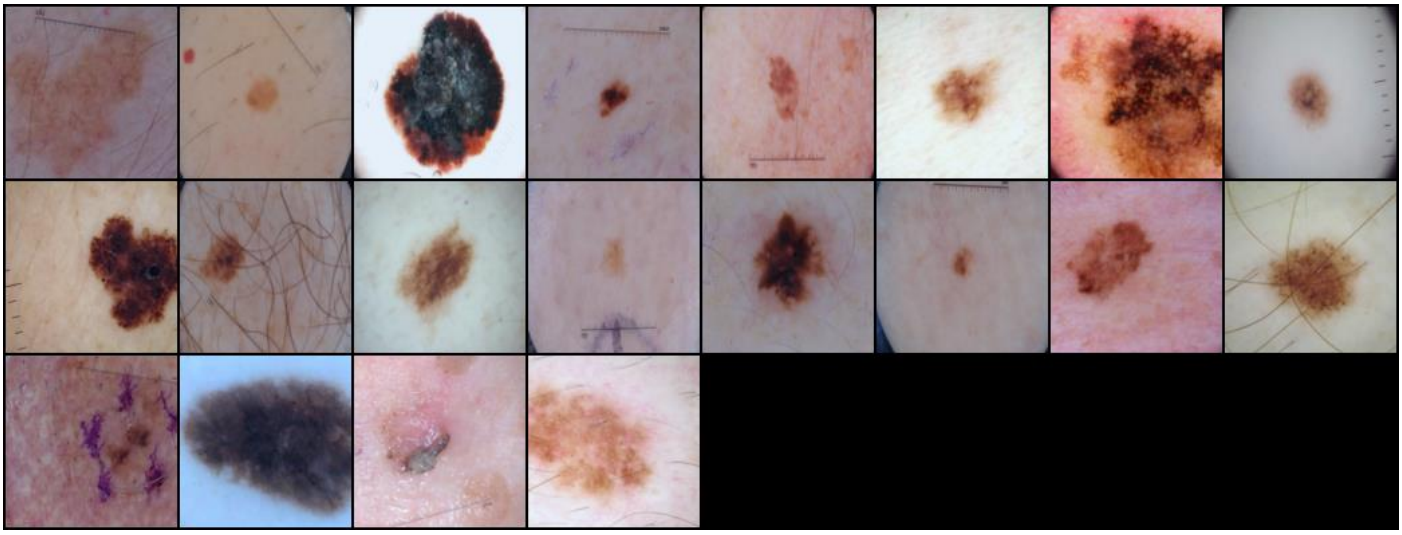
Predicted label-map by Segmentor S:



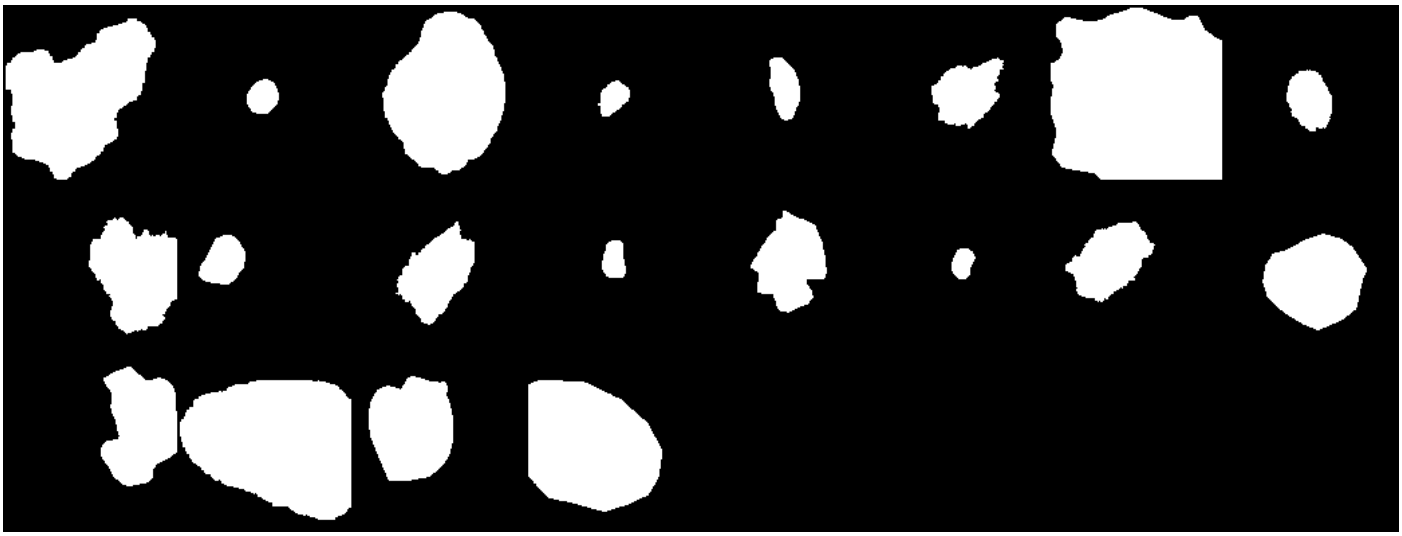
Prathyusha Akundi
2018701014

Validation result after 100th iteration:

Input:



Ground truth:



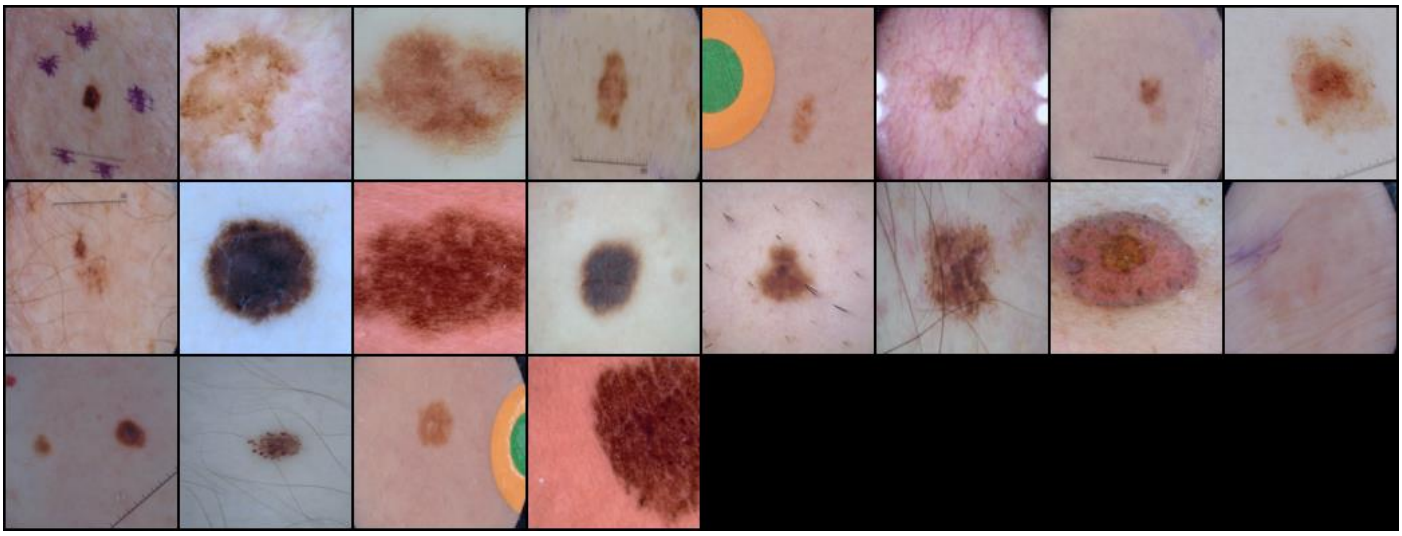
Predicted label-map by Segmentor S:

The dice coefficient for the above result is **0.9117** and IOU: **0.8503**

Prathyusha Akundi
2018701014

Iteration #1400:

Input:



Ground-truth:

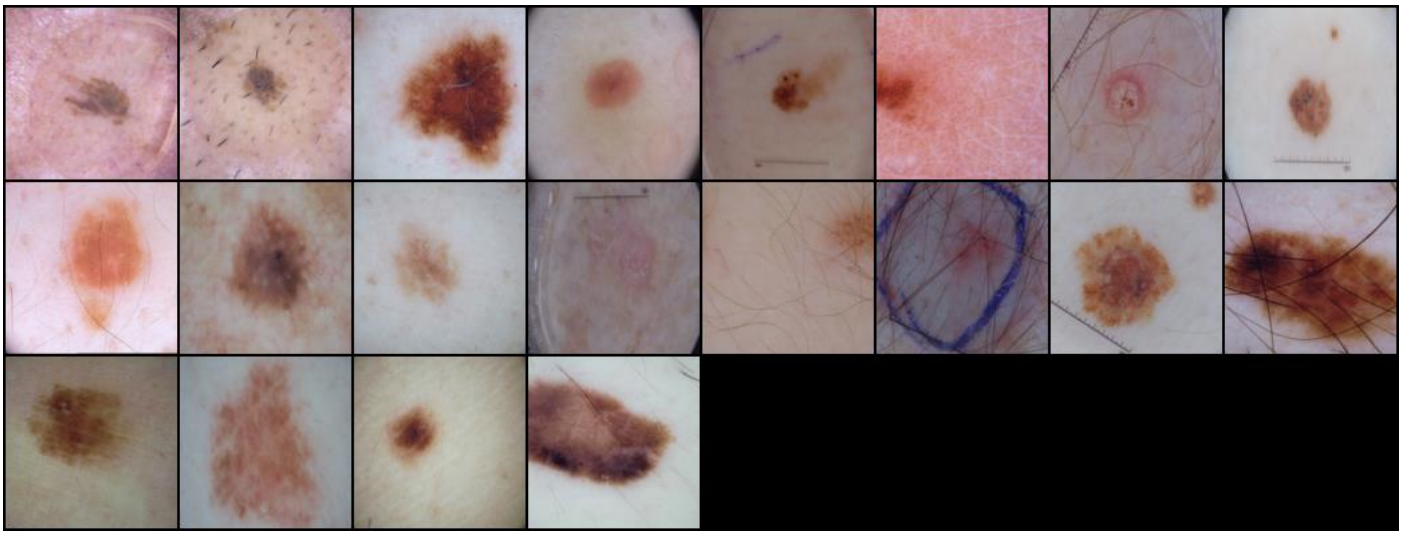
Predicted label-map:

The dice coefficient for the above result is **0.9189** and IOU: 0.8596

Prathyusha Akundi
2018701014

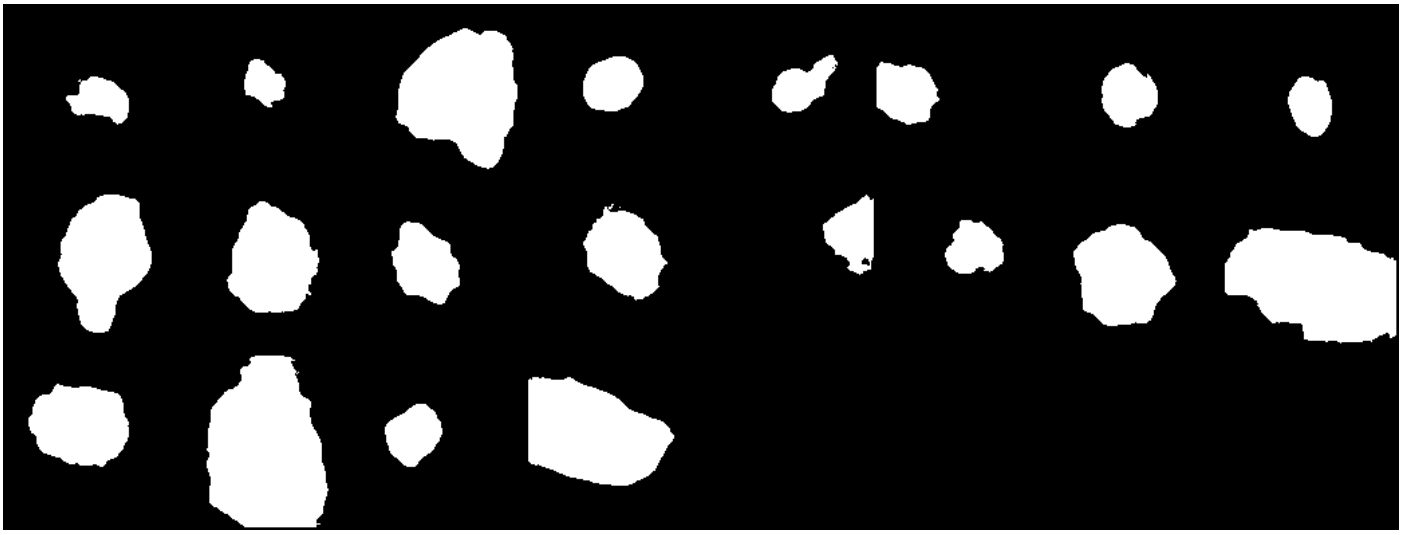
Test results:

Input:



Ground-truth:

Predicted label-map by segmentor:



The dice coefficient for the above result is 0.9189 and IOU: 0.8612

Prathyusha Akundi
2018701014

Observations:

- The main advantage of the proposed methodology is the introduction of novel segmentor-critic pair and the generalized multi-scale L1 loss function.
- Hence the segmentor will be trained according to the gradients passed through by the critic.
- But from computational complexity perspective, the proposed methodology bags certain disadvantages:
    - The above method took around 1.5 days (although the method might have converged around at 500 epochs since there is no significance change in dice coefficients from subsequent epochs) to train the network for a much simpler task like skin lesion detection.
    - Also, this only works when there is only one label map to be predicted.
    - For predicting label-maps i.e. segmented outputs for multiple labels, suppose k classes, we need to either train the network separately k times for each of these labels or create a one segmentor – k critic network or create k segmentor – 1 critic architecture.
    - This is computationally very expensive.
- For other state-of-the-art methods like U-Net, we just need to change a parameter or two to accommodate multiple label segmentation; unlike in SegAN, we have to change / recode the whole architecture of the network to perform the same task. This is the major drawback in SegAN in addition to being computationally expensive.

Conclusion:

SegAN is a novel approach to tackle segmentation problem using Adversarial Networks. With the introduction of same multi-scale L1 loss for both critic and segmentor, SegAN is able to give a descent results in segmenting skin lesions with an average dice score of about 91%. Clearly, SegAN is not limited to only a particular anatomy or in fact, this method can be applied to general semantic segmentation tasks. However, the method suffers from severe computational complexity and rigidness in employing for multiple class as we have to change the whole architecture i.e. add either segmentor or critic k times where k is number of classes. This is a major drawback as in existing methods, we just need to pass class info through the parameters during training and the whole architecture remains same. In future we can experiment to extend this work so that it generalizes well in multi-class scenario without changing the architecture everytime.