Team HolidayBurned at ISIC CHALLENGE 2018

Hao Du* Jeon Young Seok* Dianwen Ng Ngiam Kee Yuan
Mengling Feng

Saw Swee Hock School of Public Health,
National University Health System
National University of Singapore
duhao@u.nus.edu, {ephjys, ephndw, surnky, ephfm}@nus.edu.sg

* These two authors contributed equally

Introduction

Our team participated in Task 1 and Task 3 of the ISIC challenge 2018, where the former task involves image segmentation and the latter task builds machine to classify skin images to their likely diagnosed group. Since it is common for melanoma to come in huge variety of characteristics in terms of their shape, size and color along with different types of skin and texture. It is indeed very challenging for any of the tasks mentioned. To exacerbate this, some lesions appear to be irregular with fuzzy borders and low color contrast with the surrounding skin makes it even harder for the machine to perform. To combat these difficulties, we proposed a number of deep learning models to address such challenges.

Task 1: LESION BOUNDARY SEGMENTATION

A. Training Dataset

The training dataset from the internal challenge contains 2594 dermoscopic images and 2594 matching ground truth response masks [1][2]. Above that, we attempted to use additional dataset from external sources like the dermofit library. However, it did not improve the training of our model. Besides, our model with the external dataset also converged to perform slightly worse than what we had with just the internal dataset. Then, we agreed that it is likely that the ground truth masks of the image might be subjective to individuals who determine and draw the boundary. This means that the segmentation of the external sources from different person might take a different perspective in setting the boundary. Hence, our team have decided not to include dataset from other sources so that our model can focus to predict the test case of this challenge to reach a better result.

B. Methodology

We build our main model with the network architecture from the deeplab model and we employed transfer learning by taking the pre-train weight from the

PASCAL VOC- 2012. We first train our model with 20,000 iterations and then fine-tune the convolution and the decoder layers with another 20,000 iterations. In the later part, we extracted the logit features before the output layer and tried assembling with five other models trained on the same dataset. These include VGG16, U-net, DenseNet and Inception v3. The final model was ensembled using bagging method. After ensembling, we use conditional random field (CRF) to post-process the segmentation mask and obtain the final prediction for the segmentation region.

We compiled all our code with Python and conducted the experiments in Slim/Tensorflow. The experiment is done on DGX-1 machine with 8x tesla v100 and memory of 512 GB 2,133 MHz DDR4 RDIMM.

1. Image Pre-processing

Since we are not adding more data to our training sample and the volume of the internal dataset is small, we decided to execute some standard steps to image augmentation so as to increase our sample. We defined a function that listed the options of standardizing the input, vertical translation, horizontal translation, zooming in, horizontal flip and vertical flip. For each copy of the image, we pass it through the function and it will randomly take on the listed options and we will output four additional copies for each image. Then, we resize the images to 192 x 256 for training with our Deeplab model.

2. Architecture of our Network Model

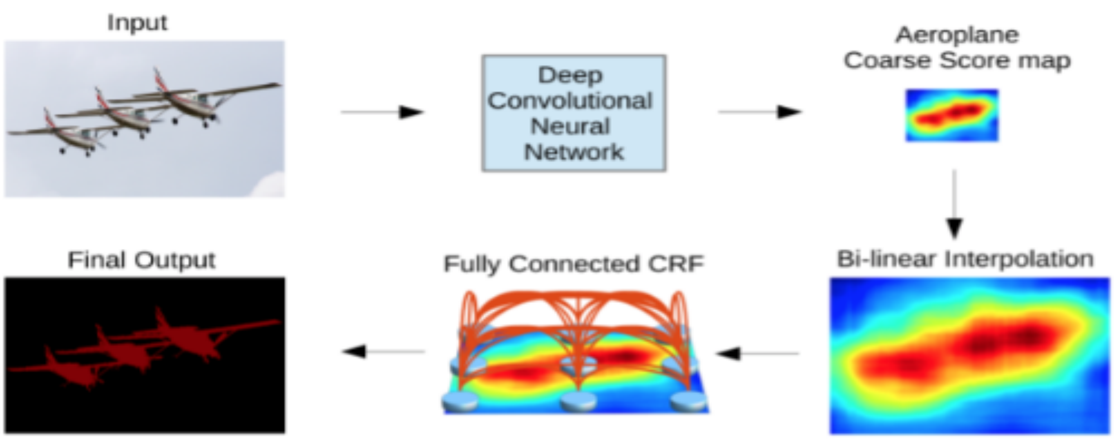

Figure 1: Network layout of Deeplab: Semantic Image Segmentation

For our main model, the backbone of this sematic image segmentation utilizes deep convolution neural network to learn the task. [3] However, it is being challenged for its reduced feature resolutions caused by the multiple max pooling and downsampling. Hence, it uses atrous convolution and bilinear interpolation (upsampling) in the last few layers to recover the original resolution of the input image. This results in more refined coarse score map. In addition, atrous spatial pyramid pooling, multiple parallel filters with rate =

6,12,18,24 is used to exploit multi-scale features. Lastly, the model uses fully connected CRF to improve the localization accuracy.

3. Results

At the last validation stage, we managed to get the result of 0.813 based on the Jaccard index on our final ensemble model.

Task 3: LESION CLASSIFICATION

A. Training Dataset

In this task, we are provided with 10,015 images and the ground truth response corresponding to each image. Our goal is to predict the confidence of the correct classes given the test images.

One of the problems we faced in this task is the allocation of the given datasets, where it is highly skewed towards certain classes. As a result, we observed sparse images for certain groups like DF, VASC and AKIEC. Since the challenge allows dataset from external sources, we planned to include more observations from the Dermofit Image Library [4]. The library contains a collection of 1,300 skin lesion images. This helps to add 66, 98, and 45 more observations to DF, VASC and AKIEC classes and of course, it increases the number of dataset we have as a whole. We will then combine all the dataset as training data and proceeds with the pre-processing phase that will be covered in the later section.

B. Methodology

The strategy we took is to assemble the softmax output predicted by three different state of the art CNN models. These three models include: Inception v4, ResNet-101 and DenseNet. We performed transfer learning by initializing the weights of some pre-train models and we run our dataset over the models for a few ten thousand iterations to fine-tune the weights so as to learn classification for the dataset of this lesion skin image. We pre-load the weights of the Inception v4 and ResNet-101 with the one released by RECOD Titan team which specifically train on the lesion image [5] and we used the weights of the ImageNet challenge to train our DenseNet.

However, note that for the last year's challenge, the model only requires to learn the classification of three classes whereas for this year, we are supposed to train our model to identify seven classes altogether. This would subsequently mean that we have to alter the output to give seven probabilities for each class.

During the training phase for our first two models, we lower the sample of the two classes appeared in the last challenge, namely benign nevi and melanomas to a size of 1000 each. Since the pre-trained weights are supposed to perform well on the two classes, by doing this, we hope to reduce over-representation of those two classes and also to provide more chances for the model to learn the other classes that are absence in the previous challenge.

We compiled all our code with Python and conducted the experiments in Slim/Tensorflow. The experiment is done on our lab server with the GPU driven by 3840 NVIDIA® CUDA® cores of 12GB GDDR5 memory.

1. Image Pre-processing

At the pre-processing stage during the training phase, we perform some simple steps that apply the same rules to each batch of 32 arbitrary selected images in one training epochs by randomly cropping a bounding box that covers the lesion area and then randomly adjusting their brightness and saturation of that cropped image. By doing so, we hope to improve the robustness of our model while preserving the original image information. Next, the image is flipped at random, either to the left or to the right side and then each channel is rescaled to [0,1]. We resize the pre-processed image to 299 x 299 to appropriately feed in the Inception model and similarly, we resize it to 224 x 224 for the ResNet and DenseNet model. During the testing phase, each case is tested independently of which, we added an additional step after the rescaling where an array of 50 patches are sampled at random from the same testing image. Our final prediction of the model will be based on the highest predicted occurrence of that array or taking average on the probability output of the 50 patches.

2. Architecture of our Network Model

The skeletons of our first two models are based on the state of the art, Inception v4 [6] and Resnet-101 [7] model. Figure 2 and 3 describes the general architecture of the Inception model and the Resnet model. The motivations behind the formal deep learning model are being very robust in analyzing image with large variation in sizes and are computationally less expensive. Furthermore, the latter model is also capable of preserving the gradient concerning the vanishing gradient problem. Hence, it seems to us as a worthy endeavor to use the two models in our work.

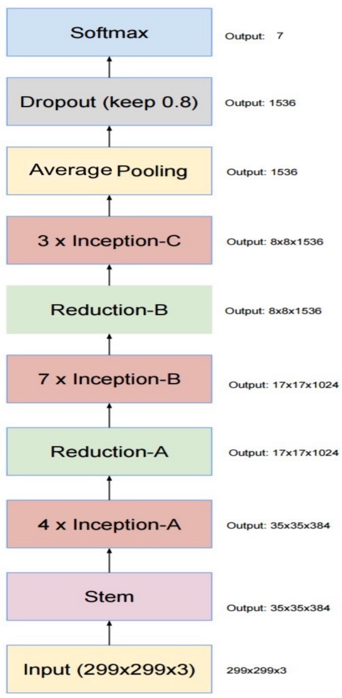

Figure 2: Network layout of Inception v4

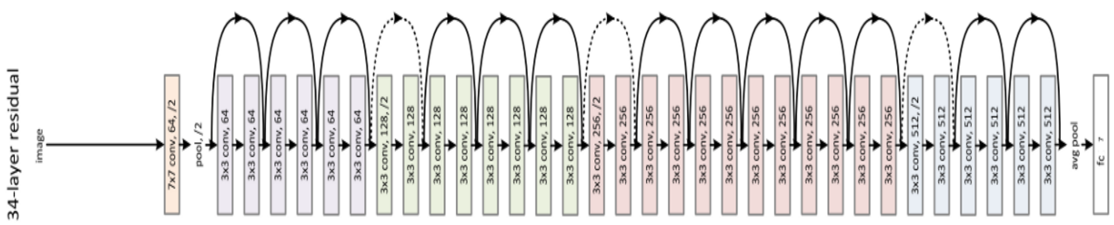

Figure 3: Network layout of ResNet

For our third model, we looked at the Dense Convolutional Network (DenseNet) proposed by G. Huang et al. 2017 [8], which claimed to outperform on most of the state of the art. We have shown the structure of the model in Figure 4 below. We wanted to highlight the two main features of a DenseNet model. First, the network is more densely coupled since each layer is connected to every other layer in a feed-forward fashion. Second, bottleneck layer is introduced to make each layer more compact and slim so as to reduce the computational cost. And in this challenge, we also applied knowledge transfer by taking the pre-train weights from the ImageNet challenge.

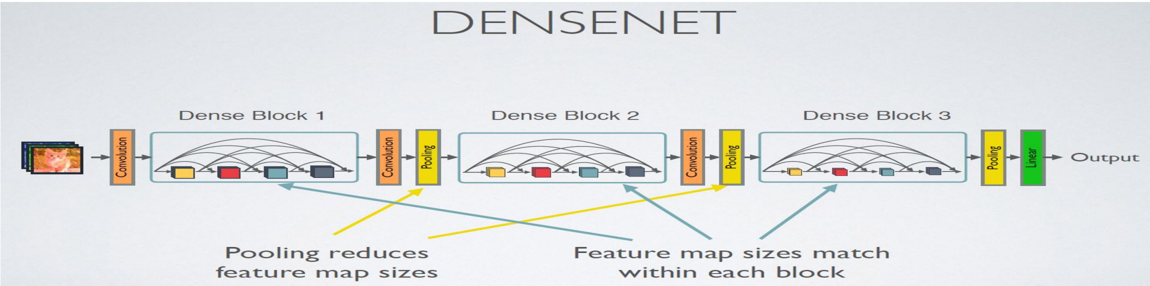

Figure 4: Network layout of DenseNet

3. Assembling the Final Submission

We trained each of our models with a maximum of 20,000 iterations. After the training is done, we take the probabilistic output of each model and then combine the results by taking the mean. We realize that it is better to set the iterations at 10,000 for the Resnet and Inception v4 model before it gets over-fitted. Unfortunately, our last model produced by the single DenseNet model did not perform well as we suspect that we only train our model on the 2018 dataset and the dermofit library. It could potentially improve if we managed to train it with the archive dataset from previous years.

4. Results

In this section, we report the score based on the score given by the validation set

Models	Validation Scores
Inception v4 (10,000 iterations)	0.874
Inception v4 (20,000 iterations)	0.870
ResNet (10,000 iterations)	0.896
ResNet (20,000 iterations)	0.868
DenseNet	0.568
Ensemble (Inception + ResNet)	0.900

Acknowledgement

Our team's GPU computation for this challenge was supported by the National Supercomputing Centre (NSCC), Singapore. Our team was also supported by the NUHS Joint Grant (R-608-000-199-733) and NUS Start-up Grant (R-608-000-172-133).

References

[1] Codella, Noel CF, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo et al. "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC)." In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, pp. 168-172. IEEE, 2018.

[2] Tschandl, Philipp, Cliff Rosendahl, and Harald Kittler. "The HAM10000 Dataset: A Large Collection of Multi-Source Dermatoscopic Images of Common Pigmented Skin Lesions." *arXiv preprint arXiv:1803.10417* (2018).

[3] Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. "Semantic image segmentation with deep convolutional nets and fully connected crfs." *arXiv preprint arXiv:1412.7062* (2014).

[4] Ballerini, Lucia, Robert B. Fisher, Ben Aldridge, and Jonathan Rees. "A color and texture based hierarchical K-NN approach to the classification of non-melanoma skin lesions." In *Color Medical Image Analysis*, pp. 63-86. Springer, Dordrecht, 2013.

[5] Menegola, Afonso, Julia Tavares, Michel Fornaciali, Lin Tzy Li, Sandra Avila, and Eduardo Valle. "RECOD titans at ISIC challenge 2017." *arXiv preprint arXiv:1703.04819* (2017)

[6] Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. "Inception-v4, inception-resnet and the impact of residual connections on learning." In *AAAI*, vol. 4, p. 12. 2017.

[7] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016.

[8] Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. "Densely Connected Convolutional Networks." In *CVPR*, vol. 1, no. 2, p. 3. 2017.