Team id	PNT2022TMID42059
Name	D.BOOBALAN
Reg.no	623119104001

Project Design Phase – I

Proposed Solution

In this paper, we present a method to sequentially combine two separate models to solve a larger problem. In the past, skin disease models have been applied to either segmentation or classification. In this study, we sequentially combine both models by using the output of a segmentation model as input to a classification model. In addition, although past studies of non-CNN segmentation models used innovative pre processing methods, recent CNN developments have focused more on the architecture of the model than on the pre processing of data. As such, we apply an innovative pre processing method to the data of our CNN segmentation model. The methods described above lack the ability to localize and classify multiple diseases within one image; however, we have developed a method to address this problem. Our objective is two-fold. First, we show that CAD can be used in the field of dermatology. Second, we show that state-of-the-art models can be used with current computing power to solve a wider range of complex problems than previously imagined. We begin by explaining the results of our experimentation, followed by a discussion of our findings, a more detailed description of our methodology, and finally, the conclusions that can be drawn from our study.

Problem Solution Fit

Analysis methodology

Our 2-phase analysis model for localization and classification is shown via the pseudocode in Algorithm. We decomposed the original image into its haemoglobin and melanin constituents using pre processing, to help our model extract valuable information from data that would have been otherwise unavailable. We provide these images as input to our segmentation model, the U-Net, which generated a segmented image. Tis segmented image was then analysed for clusters, which were subsequently cropped and input to our classification model, the Efficient Net, which then produced a classified label, thus completing our analysis model. The data for training and testing were obtained from Denet NZ, an archive of skin disease information launched and maintained by a group of dermatologists from New Zealand. The site provides open source images with labels. We selected 18 top-level categories each of which included enough data, besides including erythema as one of its common symptoms. Using a web crawler, we gathered a total of 15,851 images. Among the images obtained through Denet, the erythema of 100 images was masked by dermatologists, to be used as a ground truth. For segmentation, 60 images were used for training, and 40 images were used for testing. For classification, 13,473 images were used for training, and 2,378 images were used for testing. In addition, the test set for classification was split before segmentation cropping to prevent the subsections of one image from appearing in both the training and testing sets. Table 6 shows the distribution of data in greater detail. We chose the 100 images for segmentation in a balanced manner from each class, to minimize any bias that could occur during the classification phase. One of the significant merits of the Denet dataset is that it was created and is maintained by a diverse group of dermatologists. The images in each top-level category are independent as they are images of different patients at distinct locations taken with varying devices. This is evident in the diverse resolutions, lighting, and aspect ratios of the images. Regardless, it would be optimal to possess a similar dataset from an entirely separate association to truly validate the performance of our model.

Segmentation

The U-Net17 is an architecture created by CNNs, that has attracted attention for accurate biomedical image segmentation through the combination of down-sampling, up-sampling, and skip connections. Its name is attributed to the shape of its architecture, the first half of the ‘U’ representing down-sampling. Here, the context and key features of the input images are gained at the cost of a decrease in resolution. The second half of the ‘U’ represents up-sampling. Here, the resolution is increased to gain knowledge of the location of the target segment. To combat degradation due to the complexity of the model, skip connections are added to each up-sampling block. Although in the original paper17, the resolutions of input and output were different, that is, 572 × 572 and 388 × 388 pixels, respectively, we chose to keep our input and output resolution consistent at 304 × 304 pixels. This was done because the images in our dataset were not large enough to warrant the tiling strategy required for extremely large images. Tus, zero-padding allowed us to keep the input and output resolutions consistent, thereby allowing the retention of information present on the border of our images. Using the decomposed images, in one instance, we input three images, namely, the original, the haemoglobin, and the melanin images, to our U-Net and obtained a single black-and-white mask image as output. In this image, a black pixel represented normal skin, and a white pixel represented abnormal skin. Using the mask image, we used a simple contour-finding algorithm to draw an outline around clusters of erythema. We then used a convex hull algorithm to draw rectangles around the contours. The dimensions and locations of these rectangles were then used to crop the original image. These cropped images of each cluster were saved as individual pictures. We added padding to each cluster to create a larger and squarer image, as the performance of classification can suffer due to clusters being too small or not evenly shaped. It contours and rectangles around each cluster showing how each cluster was cropped.

Classification

EfficientNets18 were introduced in late 2019 as a state-of-the-art model for image classification. Rather than scaling a CNN model without balance between the depth, width, and resolution of the image at hand, Efficient Nets were developed by scaling a baseline model in a methodical manner. This allows for an efficient increase in accuracy rates without unreasonable amounts of required memory and floating-point operations (FLOPS) through the optimization of the formulas.

Solution Architecture