Transfer learning based framework for image segmentation using medical images and Tversky similarity

Kishore Babu Nampalle (kbabu89@cs.iitr.ac.in)
Indian Institute of Technology Roorkee
Vivek Narayan Uppala (u_vnarayan@cs.iitr.ac.in)
Indian Institute of Technology Roorkee
Balasubramanian Raman (bala@cs.iitr.ac.in)
Indian Institute of Technology Roorkee

Research Article

Keywords:

DOI: https://doi.org/

Additional Declarations: No competing interests reported.

Transfer learning based framework for image segmentation using medical images and Tversky similarity

Kishore Babu Nampalle[1*], Vivek Narayan Uppala[1] and Balasubramanian Raman[1]

*Corresponding author(s). E-mail(s): kbabu89@cs.iitr.ac.in;
Contributing authors: u_vnarayan@cs.iitr.ac.in; bala@cs.iitr.ac.in;

Abstract

Medical image segmentation and further processing are extremely difficult due to the variations in images produced by various medical imaging techniques. In addition, characteristics like color, size, shape, and the separation of the foreground and background in an image pose additional challenges. Innovative ideas and efficient designs are needed to provide more accurate results. The size of the labeled dataset also significantly impacts how effectively the proposed deep learning architecture-based model works. To achieve better segmentation results using skin cancer datasets, this paper describes a method for implementing the deep learning framework using pre-trained models. This paper proposes the MU-Net, in which the encoder part of U-Net has been replaced with a MobileNetV2 model that has already been trained and uses a focal tversky loss function to control data and class imbalances and obtain high segmentation accuracy. This method is better than existing models as it has a less number of parameters (and thus requires less computation) and is less prone to overfitting, which is a menace for medical image datasets (as most of them are small). The efficiency of the proposed work has been analyzed by comparing its performance with existing models and evaluated based on performance metrics. It surpasses the competition with high accuracy (99%) and low loss (6%). The acclaimed deep learning architecture has improved performance and is more beneficial for segmentation and classification in other domains.

Keywords: Segmentation, Transfer Learning, Skin Cancer, Deep Learning

1 Introduction

Skin cancer, melanoma, is responsible for 74% of skin disease deaths. Worldwide, its incidence is increasing faster than any other cancer, at 3-7% in many countries. Its early detection could help us decrease the mortality rate of this cancer and increase the 5-year survival rate, whose cases are on the rise [1]. Decreased diagnostic accuracy may be due to a variety of factors, including uneven and blurry lesion borders, a variety of lesion sizes, colors, and forms, and a lack of distinction between the foreground (the lesion) and the background (the surrounding skin). Consequently, there is an urgent need for computer-aided diagnostic methods for melanoma detection.

An effective and quick method to detect skin cancer is medical imaging [2]. In a broad sense, "medical imaging" refers to techniques for observing the human body to identify, track, or diagnose health disorders. Medical image analysis, which renders images more readable and enhances diagnostic efficiency, is the primary step in the processing of health data. The segmentation process is an essential phase in the evaluation of medical imagery. Segmenting the areas of medical images on which we focus and extract information is essential. This will provide a stable platform for research as well as allow clinicians to make more accurate diagnoses. Proficiently and correctly interpreting the images formed by these methods requires significant expertise in obtaining ground truth labels in healthcare applications, which is difficult [3].

Effective methods to deal with the problem of a lack of an extensive dataset include learning from computer-generated labels and learning from noisy labels. In this paper, a transfer learning technique has been employed to resolve the issue of dealing with large datasets. Any learning approach that applies the skills acquired from solving one problem to another is known as transfer learning. Transfer learning (TL) is similar to other techniques, such as multitask learning. To differentiate between transfer learning and similar related training methodologies, one issue is expected to be solved independently of another [4]. Deep learning models have been trained via transfer learning for a diverse range of medical image modeling approaches [5]. In this paper, the proposed idea employs a U-Net architecture with a pre-trained MobileNetV2 to segment skin lesion images.

We have solved all the drawbacks of previous architectures through our model, which incurs a lower computational cost (and thus requires less training time), has fewer parameters, and is less prone to overfitting in spite of the limited dataset size.

Several contributions to our paper are given below:

- The proposed model of MU-Net uses a pre-trained MobileNetV2 model and U-Net for the segmentation of the skin lesion data.

- Many of the techniques listed above are limited either by the lack of a large training dataset or by their high computational costs. These two handicaps limit the practical use of the previously used methods, at least in the near future. Overcoming these drawbacks forms the main contribution of our paper.
- We have also demonstrated our model's performance on the brain tumor dataset to further cement our MUNet's adaptability and flexibility and it's further usage for other medical image datasets.
- We used the focal tversky loss function to handle class imbalances in medical image datasets.
- We have also shown the performance of our model with and without noise.

2 Related Work

For skin lesion segmentation, numerous deep-learning architectures have been proposed. One approach uses FCN, consisting of convolution, pooling, and upsampling layers. Long et al. implemented a technique based on FCN [6], and Bi et al. suggested a multi-stage FCN for skin lesion segmentation based on the parallel integration (PI) method. This PI method is used to enhance skin lesion boundaries. When tested on the ISBI 2016 dataset, the system achieved performance with 95.6% accuracy and 91.18% dice value [7][8]. Al Masni et al. used an FCN with modification, a deep full-resolution network, to segment dermoscopy images [9]. However, FCN models are prone to over-segmentation, which can lead to crude output with insufficient training data [10].

Since U-Net was proposed in [11], there have been numerous attempts and modifications to utilize U-Net for skin lesion segmentation. In [12], they used Double U-Net, which has two U-Nets. The lesion segmentation shows that Double U-Net outperforms U-Net but this suffers from having more parameters to be trained than U-Net and thus results in a more considerable training time. SegNet is also a popular model used to segment medical images and uses an encoder-decoder model but does not employ skip connections to transmit low-level contextual data to deeper layers. It has the advantage of requiring fewer training parameters[13].

Another way is to use CDNNs [14], whose architecture consists of two major components: convolutional and deconvolutional networks. To extract distinguishing features, both networks are employed. The layers of deconvolution are used to smooth the maps of segmentation and produce high-resolution results. Yuan and Lo (2017) used this architecture (mainly using 3x3 kernels for convolution and deconvolution) on skin lesion data in various color spaces [15]. CDNNs face the same high computational cost problem as deep residual networks. The results of CDNN-based methods need to be improved, especially when the training data set is insufficient. These methods have improved

medical image segmentation and analysis performance. However, few methods among these continue to use the heavy tuning of many parameters and pre-processing methods, increasing computational cost [7][8].

3 Methodology

3.1 U-Net

A deep learning architecture called U-Net, was proposed for the segmentation of medical images [11]. It consists of an expanded path and a contracted path. The contracting portion (also known as the encoder) uses a standard CNN design. Each of its four blocks, which are composed of 3x3 convolutions as shown in equation 1, applied twice, a ReLU, a 2x2 max pooling as shown in equation 2, and a downsampling operation with stride 2, As part of the proposed work, two additional feature channels have been added for every downsampling step.

3x3 convolution:

$$y_{f,g,l} = \text{ReLU}\Big(\sum_{i\in\{-1,0,1\}}\sum_{j\in-1,0,1}\sum_{k\in\{1,\ldots,K\}} w_{i,j,k,l}x_{f+i,g+j,l+k} + b_l\Big) \quad (1)$$

2x2 max pooling:

$$y_{f,g,h} = \max_{i,j\in\{0,1\}} x_{2f+i,2g+j,h} \text{ for stride=2.} \quad (2)$$

2x2 up-convolution:

$$y_{2f+i,2g+j,h} = \text{ReLU}\Big(\sum_{h\in\{1,\ldots,K\}} w_{i,j,h,l}x_{f,g,h} + b_l\Big) \text{ for } \{i,j\} \in \{0,1\}. \quad (3)$$

The similar decoder part has four phases. The feature map is up-sampled as the very first step in the decoder path, performing a 2x2 up-convolution as shown in equation 3 to shrink the depth to half, a concatenation using the consequently cropped feature space from the encoder, and two 3x3 convolution layers, each followed by an activation function. For more accurate location information, the clipped feature maps are concatenated. In the end, we use a filter of size 1x1 to get the output segmentation map. The U-Net combines the information to obtain general information by combining the local information and context of the downsampling path with the contextual data of the upsampling path, which aids in ensuring that we predict a good segmentation map, thus bringing us the benefits of both paths. Images of different sizes can be used as input because there is no dense layer. The input ($x_{f,g,l}$) and output ($y_{f,g,h}$) pixels for a point (f, g) can be given by the equations:

$$w(f,g) = w_c(f,g) + w_0 \exp(-\frac{(d_1(f,g) + d_2(f,g))^2}{2\sigma^2}) \tag{4}$$

where $w_c(f,g)$ is a map (Eqn.4) in order to balance the class imbalances, w_0=10, $\sigma \approx 5$ pixels, $d_1(f,g)$ is distance to the nearest cell and $d_2(f,g)$ is the distance to the next nearest cell. The loss function (Eqn. 5) for U-Net is given by:

$$L = -\sum_{\mathbf{f},\mathbf{g}\in\Omega} w(\mathbf{f},\mathbf{g})\log(p_{l(\mathbf{f},\mathbf{g})}(\mathbf{f},\mathbf{g})) \tag{5}$$

where $l : \Omega \rightarrow$ {1,..., K} is the true label for all each pixels and p_k is a softmax function (Eqn. 6) applied on the final output given by:

$$p_k(\mathbf{f},\mathbf{g}) = \exp(a_k(\mathbf{f},\mathbf{g}))/\sum_{k'=1}^{K} \exp(a_{k'}(\mathbf{f},\mathbf{g})) \tag{6}$$

where a_k denotes activation of layer k [11].

3.2 MobileNet: Pre-trained deep learning architecture

3.2.1 Depth-wise separable convolution

The fundamental idea is to replace a full convolutional operator to divide convolution into two distinct layers. One convolutional filter is used for each input channel in the first layer's depth-wise convolution, which does minimal filtering. By computing linear combinations of the input channels, the 1x1 convolution, often referred to as a point-wise convolution, in the second layer generates additional features [16].

Convoluting an input of dimensions $h \times w \times d_i$ (h is the input height, w is the input width, d_i is the input depth) with n kernels of $k \times k$ dimensions, the computational cost (Eqn. 7) would be

$$Cost_{Conv} = h \times w \times d_i \times k \times k \times n \tag{7}$$

The number of parameters (Eqn. 8) required would be

$$\theta_{Conv} = k \times k \times n \times d_i \tag{8}$$

.

But with depth-wise convolution first, there will be a $k \times k$ kernel per input channel and then stacked on top of each, so it will cost (Eqn. 9) only

$$Cost_{DWConv} = h \times w \times d_i \times k \times k \tag{9}$$

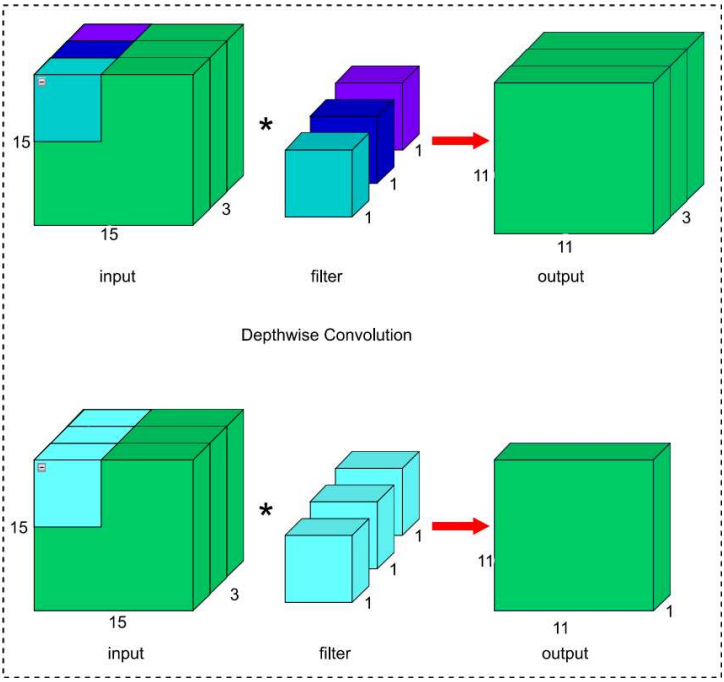


Fig. 1: Depth-wise convolution vs Standard convolution

The number of parameters (Eqn. 10) required here would be

$$\theta_{DWConv} = k \times k \times d_i \tag{10}$$

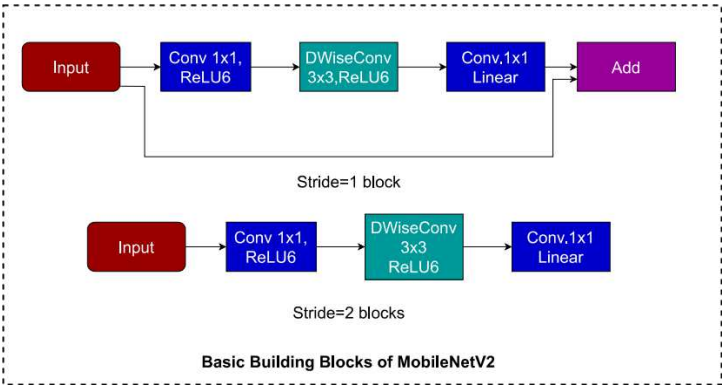


Fig. 2: MobileNetV2 Model

The second step would involve point-wise convolution, that is convolution with n 1x1 filters(this would be like a standard convolution) with a cost (Eqn. 11) of

$$Cost_{PWiseconv} = h \times w \times i \times n \tag{11}$$

Here, the number of parameters (Eqn. 12) required would be

$$\theta_{PWiseconv} = d_i \times n \tag{12}$$

Hence, the total computational cost (Eqn. 13) for Depth-wise convolution is,

$$Cost_{DWSepconv} = h \times w \times d_i \times (k \times k + n) \tag{13}$$

So, for depth-wise separable convolution, the cost (Eqn. 14) would be $k \times k(k^2)$ times smaller than for a typical convolution.

$$\theta_{DWSepconv} = d_i(k \times k + n) \tag{14}$$

Hence, the parameters are approximately n times less than the standard convolution for depth-wise separable convolution. Here, in MobileNetV2, we use bottleneck depth-wise separable convolutions, which consist of a convolutional layer with 1x1 kernels and ReLU6 (expansion layer), then a depth-wise convolution with ReLU6 and, at last, a 1x1 kernel convolution with a linear function. Also, we add the previous layer's output to the output of this layer(shortcut) in an inverted fashion(inverted residuals) [16] [17].

3.3 Proposed methodology: U-Net with MobileNetV2(MU-Net)

We know that U-Net has been effective in segmenting images. Still, we have seen that large datasets for training the U-Net network are unavailable, reducing the U-Net's accuracy. So, here we use the approach of Transfer Learning [18]. Pre-trained architectures, trained on a big dataset to find a solution for the problem, can be used for other similar problems with the same weights and architecture. This is known as transfer learning. Many pre-trained models are available [19].

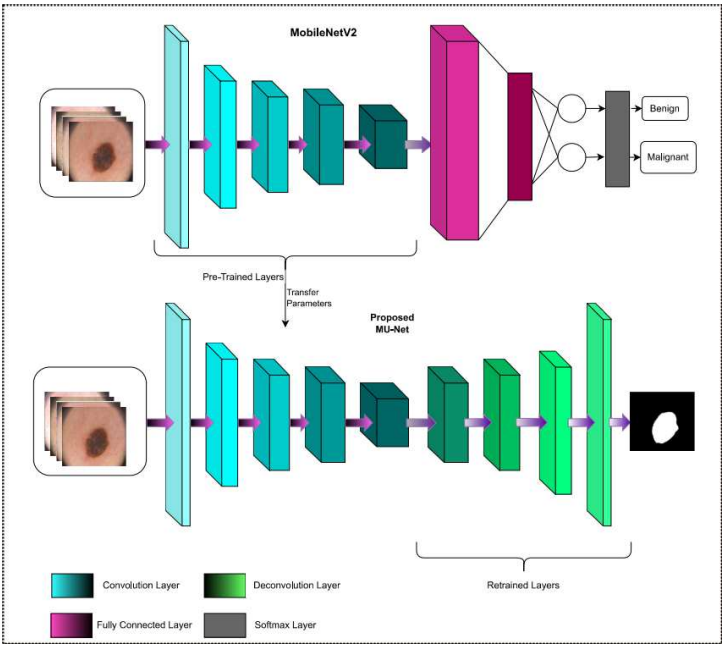


Fig. 3: The proposed MU-Net and MobileNetV2.

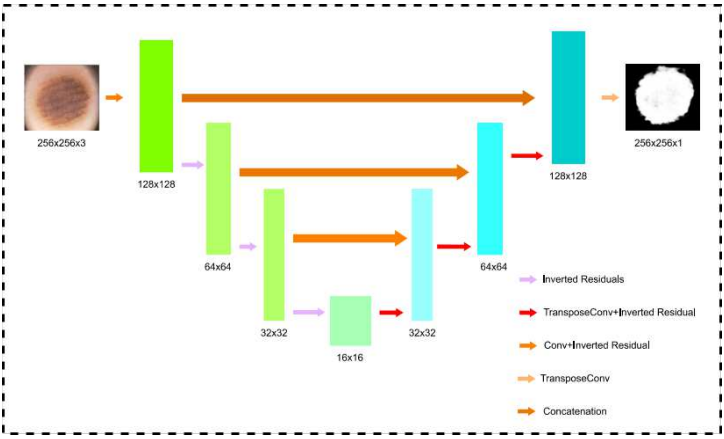


Fig. 4: The proposed architecture of MU-Net with a U-Net backbone and Pre-trained MobileNetV2 replacing the Downsampling path of U-Net .

But, Compared to other pre-trained models, MobileNetV2 has fewer parameters to train. It also provides a high level of accuracy and uses little computing power and storage. Thus, a U-Net model with a modified encoder, MobileNetV2, outperforms a U-Net model in terms of segmentation accuracy.

As the features extracted in the decoding part in U- Net are high-level features, the more complex network structure requires a lot of parameters and memory and predicts slowly. MobileNetV2 is quite the opposite of this in the sense that it requires fewer parameters, less memory, and predicts faster. So, in the proposed methodology, the encoder component of U-Net is swapped out with a pre-trained MobileNetV2 model, which is then utilized to segment the images from the datasets for skin lesions and brain tumors. Also, a point to note is that the pooling layers used in U-Net are not used in our proposed MU-Net.

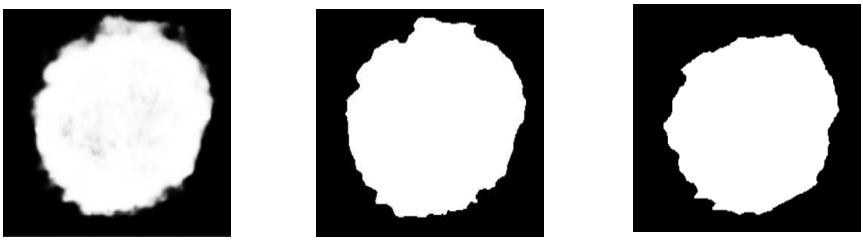
The faster, less computationally complex proposed model has the potential for deployment to segment skin lesion images. The input image is a 256x256 image with 3 channels (RGB). Pre-trained MobileNetV2 has been used as a backbone for the encoder. The first block changes the shape to 128x128 with 144 feature channels. Then taking this as input, the second block changes the size to 64x64 with 192 feature channels. Then, the next block changes the size to 32x32 with 288 channels. Then, the last encoding block changes the size to 16x16 with 816 channels. Then, we come to the upsampling path or decoder part of U-Net. In this part, the input is deconvoluted and then concatenated with corresponding same-sized outputs from the encoder part, and at last, a deconvolution layer is used to obtain the predicted mask. The predicted mask is of size 256x256 with 1 channel (grayscale).

3.3.1 Loss Function

We used the focal tversky loss function for training our model. It is similar to focal loss as it involves focusing on hard examples by weighing down easy examples [20]. The dice function weighs FP and FN equally. Hence it leads to high precision and low recall. The FN detections need to be weighted higher than FPs to improve the recall rate. The inability of DL for segmenting the small ROIs, which do not significantly contribute to the loss, is another area for development. To address this issue, focal tversky is used as a loss function in which there is flexibility in weights of FP and FN [21]. The focal tversky loss function solves the data imbalances frequently occurring in Medical Image Segmentation datasets better than Dice loss, focal loss or any cross entropy based loss functions. Also, using this loss helps us achieve the best trade-off between precision and recall when compared to the other functions.

3.3.2 Post Processing

As a result of pixel values being between the range 0-1, the images are blurry, and the edges are unclear. Therefore, we use a final enhancement technique to distinguish the boundaries by rounding up values greater than 0.5 to 1 and the rest to 0. This way, we can get clear edge distinction, and this also helps in improving accuracy. The images before and after post-processing for a particular skin lesion image are shown below.



Predicted mask Enhanced results Original mask

Fig. 5: Samples of images before and after post-processing.

4 Dataset and Pre-Processing

4.1 Dataset

For segmenting lesion parts, the ISIC 2019 dataset was used, which consists of 33,569 total images 25,331 for training and 8238 for testing. Only the training dataset has access to the ground truth data, which includes the several classes [22][8]. This dataset is publicly available on Kaggle.

The ISIC 2019 dataset is noteworthy since it includes multiplets of single lesions that is they contain the same image at various magnifications, which offer unique crucial features [22]. It is taken from BCN20000 [23], HAM10000 [24], and MSK datasets [25][26][8]. Out of the 33,569 images, we have trained 600 images for our network.

The brain tumor dataset is obtained from Kaggle, and it comprises of three datasets: figshare [27], SARTAJ dataset, and Br35H dataset[28]. This dataset consists 7022 human brain MRI images divided into four classes. The brain no-tumor class images have been procured from the Br35H dataset. Also, in the SARTAJ dataset, there is a problem, the glioma class images need to be categorized correctly. So those images are deleted in this folder, and we used the images on the figshare site instead. Of the 7022 images, 1311 are utilized for testing, and 5711 images are utilized to train the model.

4.2 Pre-processing

As discussed previously, the data available for medical imaging is relatively limited. Various data augmentation techniques are employed to augment medical data. Using a variety of processing techniques, image augmentation artificially builds training pictures.

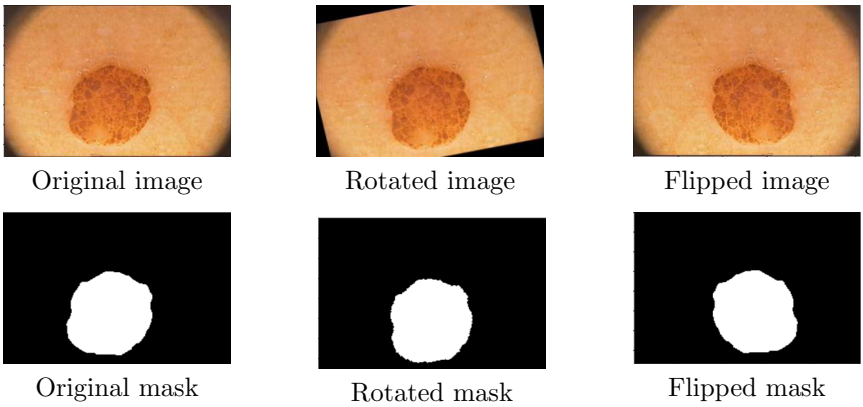


Fig. 6: Samples of images before and after pre-processing.

We can also use combinations of several processing techniques, such as random rotation, shear, shifts, flips, etc. In order to increase the data available for training the model, we applied two methods, namely random rotations and horizontal flips.

5 Experiments

5.1 Setup

We implemented the network using Python with the Tensorflow and Keras libraries. These experiments were run on an Intel Core i5-9300H processor with a primary frequency of 2.40 GHz, 8GB RAM, NVIDIA GTX 1050 Graphics card. The operating system used is the 64-bit Windows 10 Home Operating System. The model is trained in 60 epochs. We have utilized the Adam with a learning rate of 0.05.

5.2 Metrics

For the accuracy of the proposed network, six metrics have been used, namely Tversky, IOU, Dice Coefficient, Precision, Recall, and Accuracy.

- **Accuracy:** When discussing image segmentation, accuracy is the proportion of successfully segmented pixels to all of the image's pixels. Four parameters can be used to define accuracy: TN, TP, FN, FP.
  - TP: It shows the number of pixels accurately identified as nucleus pixels. TP can be written as AA_{pr}.
  - TN: The number of backdrop pixels that were accurately anticipated. TN can be written as $\bar{A}\bar{A}_{\text{pr}}$.
  - FP: Number of nucleus pixels mistakenly projected to be background pixels. FP can be written as $\bar{A}A_{\text{pr}}$.

– FN: Nuclei pixels that were mistakenly projected to be background pixels in quantity. FN can be written as $A\bar{A}_{\text{pr}}$.

The formula for Accuracy is given by equation [15].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

A more accurate result is the one that is closer to 1.

- **Tversky Index:** The Tversky Index is a commonly used metric for medical image segmentation where accuracy is unreliable as imbalanced datasets occur frequently and the classes don't have equal importance. For example, it is more crucial to correctly identify malignant regions than healthy ones when it comes to cancer detection. This is even though these regions often include a relatively small amount of tissue. The smoothed Tversky Index (16) used in this paper is given by:

$$TI = \frac{TP + \delta}{TP + \alpha FP + \beta FN + \delta} \tag{16}$$

where δ represents a small smoothing term(hence this index is called smoothed Tversky index) to ensure the Tversky Index is continuous and α and β are adjustable parameters to enhance the value of the recall rate. In this paper, we have taken α as 0.7 and β as 0.3. The lower value of β regulates the FP [21], whereas the greater value of α improvises the network to reduce the FN. Many problems in DL are prone to ill-conditioning when the output of their final layer contains either extremely large or extremely small coordinates. Using a softmax or sigmoid activation function, yields predictions that are quite near to 0 or 1. Since the gradient of the loss function tends to 0 in infinities, this might lead to overfitting and reduce the model's adaptability because weight updates made during training will have lower magnitudes. So, we introduce a method called label smoothing (17) where we replace A with

$$A_{new} = A(1 - \epsilon) + (\epsilon)1 \tag{17}$$

The value of epsilon is considered as e^{-5} in our work.

- **IOU(Intersection over Union):** The similarity of the segmented outcome to the original mask of the pathologist is what is meant by this term. Assume that A represents the image of the ground truth mask and B represents the output of the model. Calculating IOUs can be done using equation 18. The IOU value is between 0 and 1. The segmentation model has been properly trained and is suitable for segmentation when the output and the mask are similar, as shown by a high IOU value. The IOU or Jaccard Index is just the Tversky Index with α and β both as 1.

$$\textbf{IOU}(A,B) = \frac{A \cap B}{(A+B)-(A \cap B)} = \frac{TP}{TP+FP+FN} \tag{18}$$

- **Dice coefficient:** Positives are not the only thing we find out, we also find out the penalties for false positive predictions. Precision rather than accuracy is the result of this feature. The formula for the dice coefficient is given in equation 19. The Dice Coefficient is nothing but the Normal Tversky Index with α and β both equal to 0.5.

$$\textbf{Dice}(A,\ B) = 2\times\frac{\mathsf{A}\cap\mathsf{B}}{\mathsf{A}+\mathsf{B}} = \frac{2\times TP}{2\times(TP)+FP+FN} \tag{19}$$

- **Precision:** It (Eqn. 20) calculates the proportion of accurately identified pixels:

$$\textbf{Precision}(A,B) = \frac{A \cap B}{B} = \frac{TP}{TP+FP} \tag{20}$$

- **Recall:** It (Eqn. 21) gauges the proportion of actual detection of ground truth positives:

$$\textbf{Recall}(A,B) = \frac{A \cap B}{A} = \frac{TP}{TP+FN} \tag{21}$$

5.3 Ablation Study

5.3.1 Effect of using pre-trained Model

Using transfer learning helped reduce the training time as weights were initialized from training on a larger dataset available in the TensorFlow library. It also had a profound impact on the overall score because the model did not overfit in spite of a small dataset.

5.3.2 Effect of using MobileNetV2

From the results, it's clear that although transfer learning had a part, the model which was used as the encoder part still had a role to play. Although the residual connections of ResNet50 improve performance the inverted residuals and using ReLU6 instead of ReLU to expand upon the output space thus outperform ResNet50. Pointwise convolutions while being powerful tools to reduce computation restrict the extraction of features because they can just take the linear combinations of feature map inputs and using n such kernels on the same input doesn't extract as many features as n 3x3 kernels and thus is less flexible. But, here the linear bottlenecks and inverted residuals clearly turn the game in favor of MobileNetV2.

5.3.3 Effect of using focal tversky loss

Focal tversky loss helped improve the convergence of the model to a better optimum and thus enhanced the results we got. This was also observed in the case of other models used as a comparison too. This handles data imbalance well and provides a better trade-off between precision and recall. This especially works well with Medical image datasets where data imbalances are frequent

5.3.4 Effect of using Pre Processing and Post Processing

Usage of various Pre Processing and Post-processing techniques helped us achieve improved results.

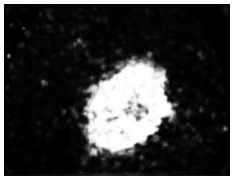
5.3.5 Effect of noise

We have examined the model's performance with and without noise. This was done as, in real-world applications, noise is frequently encountered, and we wished to test our model's performance in its presence.

5.4 Qualitative results analysis

(a) The input image.

(b) The actual segmented mask.



(c) Output predicted by U-Net with VGG-19.

(d) Output predicted by the proposed model.

Fig. 7: Comparison of outputs

In demonstrating the results of the proposed MU-Net model, the comparison has been done with various DL-based image segmentation models like FCNet, U-Net with VGG-19, SegNet, and U-Net with ResNet-50. All the methods are evaluated using the skin lesion and brain tumor datasets. We have also included the results of our model with and without noise.

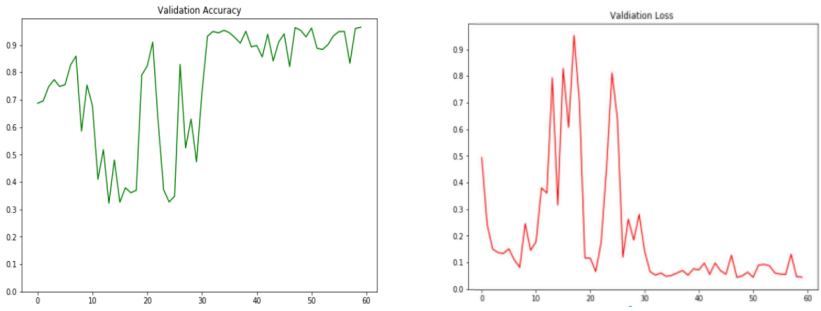


Fig. 8: Result analysis of the proposed work using metrics accuracy and Loss function on skin lesion dataset.

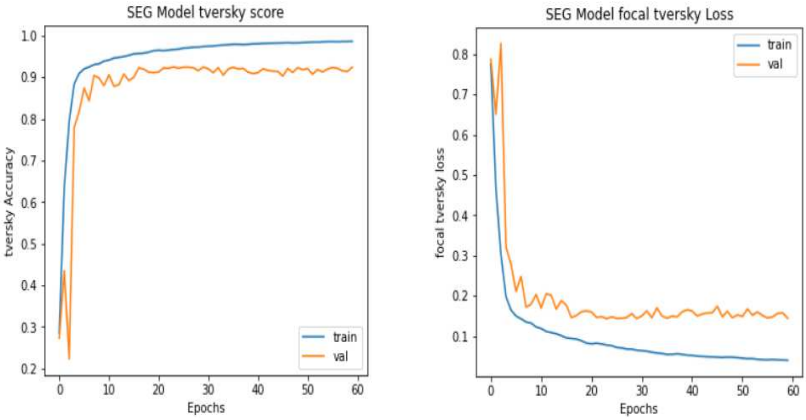


Fig. 9: Result analysis of the proposed work using metrics accuracy and Loss function on brain tumor dataset.

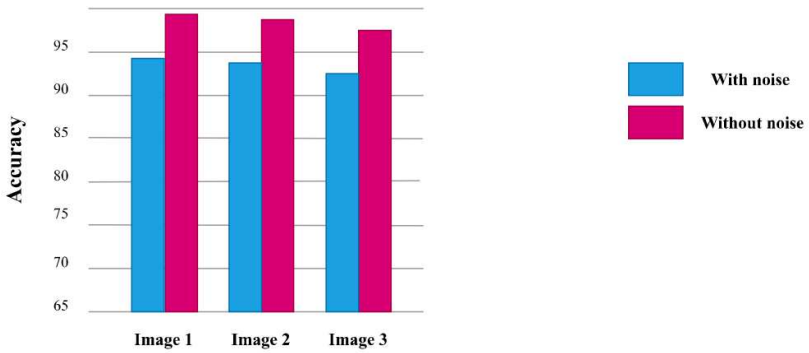
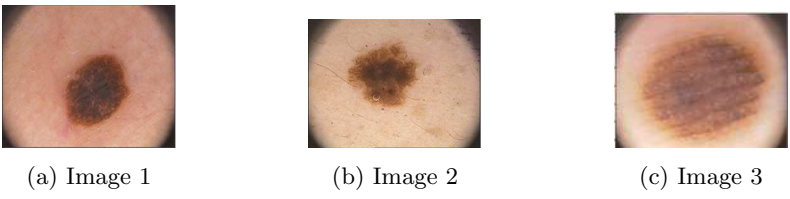
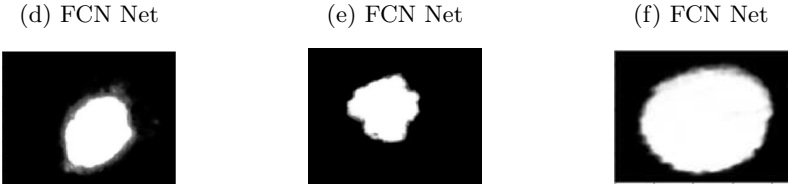


Fig. 10: Performance of a model with noise and without noise

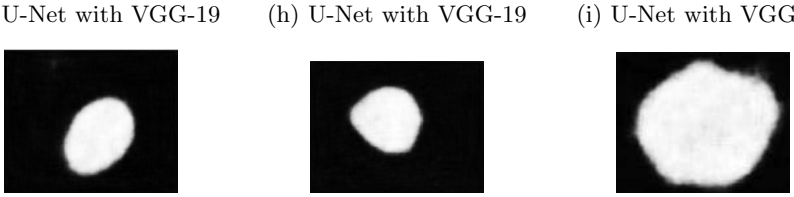


(a) Image 1 (b) Image 2 (c) Image 3

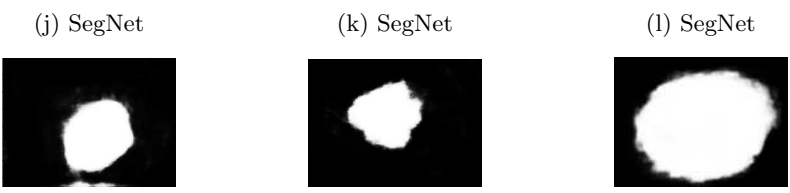
(d) FCN Net (e) FCN Net (f) FCN Net



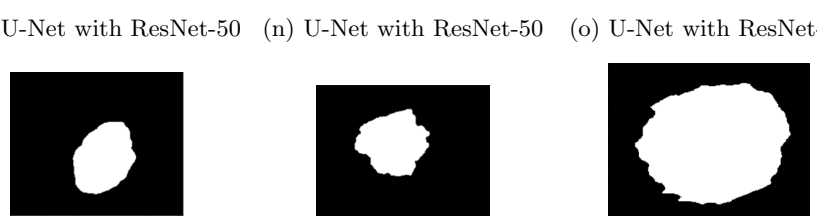
(g) U-Net with VGG-19 (h) U-Net with VGG-19 (i) U-Net with VGG-19



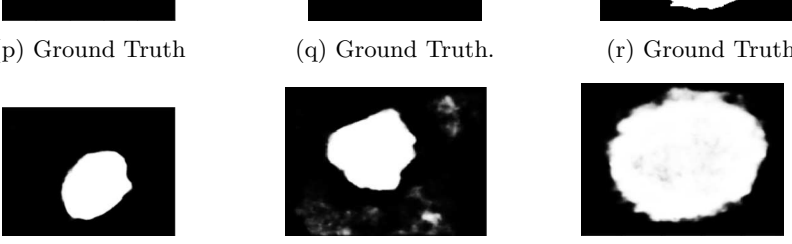
(j) SegNet (k) SegNet (l) SegNet



(m) U-Net with ResNet-50 (n) U-Net with ResNet-50 (o) U-Net with ResNet-50

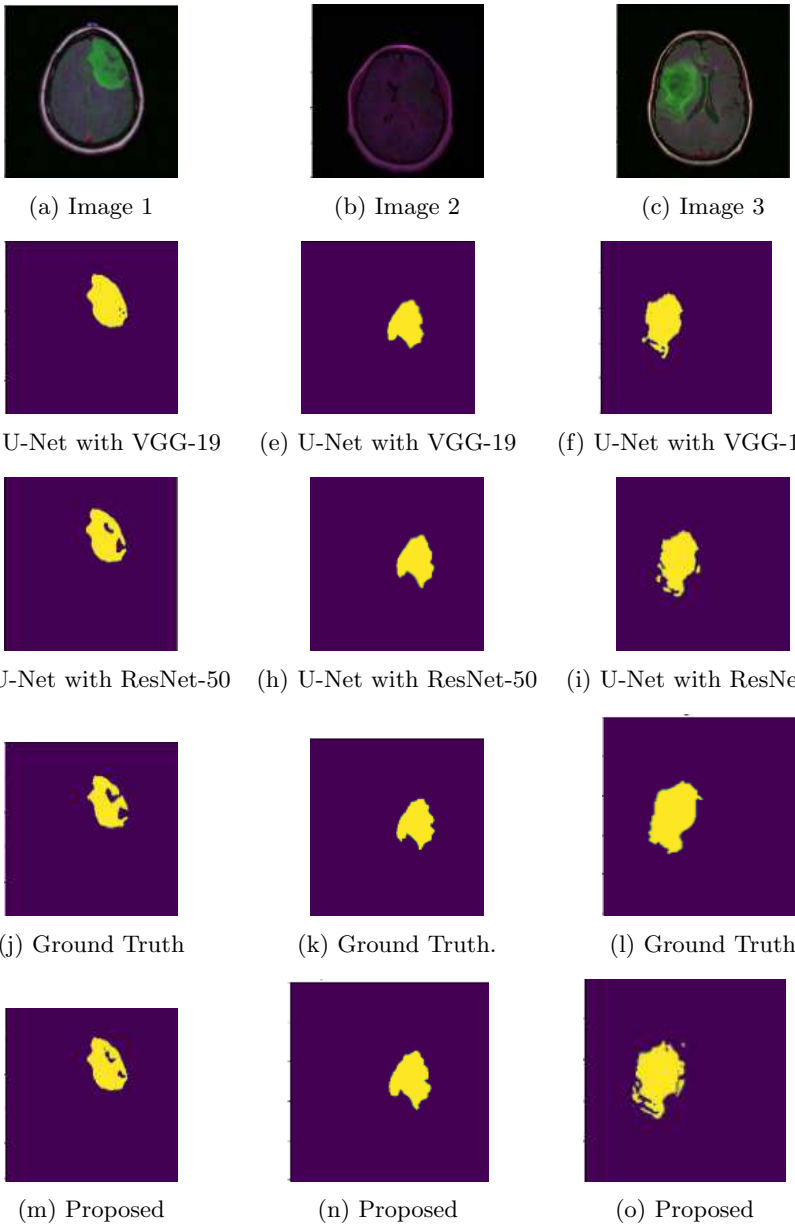


(p) Ground Truth (q) Ground Truth. (r) Ground Truth



(s) Proposed (t) Proposed (u) Proposed

Fig. 11: Qualitative analysis of results of the proposed method using skin imagery.



(a) Image 1 (b) Image 2 (c) Image 3

(d) U-Net with VGG-19 (e) U-Net with VGG-19 (f) U-Net with VGG-19

(g) U-Net with ResNet-50 (h) U-Net with ResNet-50 (i) U-Net with ResNet-50

(j) Ground Truth (k) Ground Truth. (l) Ground Truth

(m) Proposed (n) Proposed (o) Proposed

Fig. 12: Qualitative analysis of results of proposed method brain imagery.

5.5 Quantitative results analysis

We use the quantitative metrics discussed in Section 5.2, namely accuracy, recall, precision, IOU, Tversky, and the Dice Coefficient, to quantitatively

analyze the performance of various deep learning-based architectures for skin lesion segmentation. The experimental results on the skin lesion dataset are shown in table 1. The proposed model either closely competes with the most effective model for that metric or surpasses the existing techniques for segmentation tasks.

Table 1: Performance comparison based on the metrics discussed in Section 5.2 on the skin lesion dataset.

Model	Accuracy	Recall	Precision	IOU	Tversky	Dice Coef
U-Net with VGG-19	88.94	70.82	92.06	89.92	72.77	75.57
SegNet	95.46	94.28	91.45	95.03	84.08	82.17
U-Net with ResNet-50	95.35	95.92	89.92	95.64	92.50	91.07
FCNet	89.51	72.04	93.45	88.36	67.26	65.81
U-Net-224	94.66	87.50	**95.37**	95.64	88.79	90.13
Proposed	**96.49**	**97.99**	91.44	**96.42**	**95.16**	**93.78**

Table 2: Performance comparison based on the metrics discussed in Section 5.2 on brain tumor dataset.

Model	Accuracy	Recall	Precision	IOU	Tversky	Dice Coef
U-Net with VGG-19	**99.45**	88.51	**90.86**	**99.99**	89.13	89.60
SegNet	98.69	73.45	86.75	97.71	53.40	45.85
U-Net with ResNet-50	99.37	90.02	87.74	**99.99**	89.26	88.81
FCNet	96.01	**99.56**	46.90	93.82	69.84	58.40
U-Net-224	99.15	89.88	80.91	98.42	86.92	85.11
Proposed	99.40	93.10	90.06	99.06	**92.14**	**91.54**

6 Discussion

The proposed MU-Net achieves outstanding performance on the skin lesion dataset, with only marginally lower precision than U-Net-224. MU-Net requires fewer epochs to reach convergence, which shows it has a robust architecture. U-Net with ResNet-50 performs well (albeit not as well as the proposed model). This is due to the presence of residual networks. It is essential to note that we only use 11,753,809 total parameters, which is much lower than in the existing models. The second lowest number of total parameters is 25,751,426 from U-Net with ResNet-50. To understand the impact of the proposed model, this number is 2.19 times the required parameters of the proposed model and still exhibits comparable or better performance. We also see that the model is adaptable and flexible. Despite certain drawbacks, the model's performance makes it very useful and presents a convincing picture of how deep learning can change the face of medical imaging.

7 Conclusion and future work

In this paper, a novel and efficient U-Net-based model for skin lesion image segmentation has been proposed. The proposed model overcomes the limitation of limited training datasets for medical image segmentation by utilizing MobileNetV2 trained on large datasets for the encoder part of U-Net. Through transfer learning, we also avoid the common drawback of overfitting. Image augmentation is used to enlarge the limited datasets. The effectiveness of the proposed work is tested on the skin lesion and brain tumor datasets through evaluation of IoU, Dice coefficient, accuracy, recall, precision, and Tversky metrics. The results are compared with various models. The initial results are promising and outperform different state-of-the-art techniques and other models. Its utility is more comprehensive than the skin lesion dataset alone, especially when training data is limited or computational constraints exist. It's excellent performance on brain tumor images proves it. In the future, additional techniques can be added to improve outcomes, including pre-trained models (like MobileNetV3) and segmentation designs (using U-Net++) on other medical images. Post-processing techniques can also be employed to enhance the results and improve accuracy.

Our paper all in all demonstrates superior accuracy with lesser computation. Our work presents a real opportunity for deployment in Medical field because of the drastic time reduction in the prediction of results. Current processing technology also facilitates swift results using our method.

Acknowledgments

The research was conducted in the Machine Intelligence Lab, Indian Institute of Technology Roorkee and was funded by the Ministry of Education, Government of India, using a grant from IIT Roorkee (Grant No: OH-31-24-200-428).

Disclosure statement

The authors did not disclose any apparent conflicts of interest.

Ethical Approval

This article does not contain any of the authors' research involving humans or animals.

References

[1] Howard, M.D.: Focus: Skin: Melanoma radiological surveillance: A review of current evidence and clinical challenges. The Yale Journal of Biology

and Medicine **93**(1), 207 (2020)

[2] Das, S., Nayak, G.K., Saba, L., Kalra, M., Suri, J.S., Saxena, S.: An artificial intelligence framework and its bias for brain tumor segmentation: A narrative review. Computers in Biology and Medicine, 105273 (2022)

[3] Aljabri, M., AlAmir, M., AlGhamdi, M., Abdel-Mottaleb, M., Collado-Mesa, F.: Towards a better understanding of annotation tools for medical imaging: a survey. Multimedia Tools and Applications, 1–35 (2022)

[4] Karimi, D., Warfield, S.K., Gholipour, A.: Transfer learning in medical image segmentation: New insights from analysis of the dynamics of model parameters and learned representations. Artificial Intelligence in Medicine **116**, 102078 (2021)

[5] Cheplygina, V., de Bruijne, M., Pluim, J.P.: Not-so-supervised: a survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. Medical image analysis **54**, 280–296 (2019)

[6] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)

[7] Bi, L., Kim, J., Ahn, E., Kumar, A., Fulham, M., Feng, D.: Dermoscopic image segmentation via multistage fully convolutional networks. IEEE Transactions on Biomedical Engineering **64**(9), 2065–2074 (2017)

[8] Adegun, A., Viriri, S.: Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of state-of-the-art. Artificial Intelligence Review **54**(2), 811–841 (2021)

[9] Al-Masni, M.A., Al-Antari, M.A., Choi, M.-T., Han, S.-M., Kim, T.-S.: Skin lesion segmentation in dermoscopy images via deep full resolution convolutional networks. Computer methods and programs in biomedicine **162**, 221–231 (2018)

[10] Ramachandram, D., DeVries, T.: Lesionseg: semantic segmentation of skin lesions using deep convolutional neural network. arXiv preprint arXiv:1703.03372 (2017)

[11] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 234–241 (2015). Springer

[12] Jha, D., Riegler, M.A., Johansen, D., Halvorsen, P., Johansen, H.D.:

Doubleu-net: A deep convolutional neural network for medical image segmentation. In: 2020 IEEE 33rd International Symposium on Computer-based Medical Systems (CBMS), pp. 558–564 (2020). IEEE

[13] Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(12), 2481–2495 (2017)

[14] Ashraf, H., Waris, A., Ghafoor, M.F., Gilani, S.O., Niazi, I.K.: Melanoma segmentation using deep learning with test-time augmentations and conditional random fields. Scientific Reports **12**(1), 3948 (2022)

[15] Yuan, Y., Lo, Y.-C.: Improving dermoscopic image segmentation with enhanced convolutional-deconvolutional networks. IEEE journal of biomedical and health informatics **23**(2), 519–526 (2017)

[16] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)

[17] Han, D., Kim, J., Kim, J.: Deep pyramidal residual networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5927–5935 (2017)

[18] Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big data **3**(1), 1–40 (2016)

[19] Minaee, S., Boykov, Y.Y., Porikli, F., Plaza, A.J., Kehtarnavaz, N., Terzopoulos, D.: Image segmentation using deep learning: A survey. IEEE transactions on pattern analysis and machine intelligence (2021)

[20] Jadon, S.: A survey of loss functions for semantic segmentation. In: 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), pp. 1–7 (2020). IEEE

[21] Abraham, N., Khan, N.M.: A novel focal tversky loss function with improved attention u-net for lesion segmentation. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pp. 683–687 (2019). IEEE

[22] Cassidy, B., Kendrick, C., Brodzicki, A., Jaworek-Korjakowska, J., Yap, M.H.: Analysis of the isic image datasets: usage, benchmarks and recommendations. Medical Image Analysis **75**, 102305 (2022)

[23] Combalia, M., Codella, N.C., Rotemberg, V., Helba, B., Vilaplana,

V., Reiter, O., Carrera, C., Barreiro, A., Halpern, A.C., Puig, S., et al.: Bcn20000: Dermoscopic lesions in the wild. arXiv preprint arXiv:1908.02288 (2019)

[24] Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific data **5**(1), 1–9 (2018)

[25] Codella, N., Rotemberg, V., Tschandl, P., Celebi, M.E., Dusza, S., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M., et al.: Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). arXiv preprint arXiv:1902.03368 (2019)

[26] Codella, N.C., Gutman, D., Celebi, M.E., Helba, B., Marchetti, M.A., Dusza, S.W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., *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: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), pp. 168–172 (2018). IEEE

[27] Cheng, J.: Brain tumor dataset. figshare. dataset. ed (2017)

[28] Hamada, A.: Br35h:: Brain tumor detection 2020. Kaggle (2020)