MELANOMA IMAGE SEGMENTATION USING SELF ORGANIZED FEATURE MAPS

Anirudh Munnangi[1], **Prabir Bhattacharya**[2]
[1]Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, OH, USA
[2]Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, OH, USA

Abstract *- This paper presents a SOFM (Self Organizing Feature Maps) model addressing the problem of segmentation of Dermoscopic skin cancer images. It proposes a unique way of passing information from the image to the network and shows how to interpret the output of the network. The main aim is to train the network so that it segments novel images correctly. The performance has been compared with standard existing methods and relevant comparative observations have been made. Experimental testing has been done on 420 Dermoscopic images which demonstrate the effectiveness of the model.*

Keywords: Dermoscopy, Kohonen networks, Image processing, Melanoma, Segmentation.

1 Introduction

Skin cancer (melanoma) detection is one of the most challenging problems faced by mankind. However, as per research, it is proven that it can be cured given its early and accurate detection. One of the established ways to perform the detection is through Dermoscopy. Dermoscopy is a non-invasive diagnostic technique for the in vivo observation of pigmented skin lesions used in dermatology [1]. Dermoscopy uses tools like non-polarized light, coupled with liquid medium and a transparent plate so as to get an effective image. The role of the specialist is to check these images and give the opinion regarding the status of the cancer. One major drawback of human based inference is that it subjects the detection process to human error based on the skills and experience of the person performing the diagnosis. So, it becomes prudent to have a second opinion which is given by the automated computerized detection approach [1].
The process of skin cancer detection is done stepwise and the individual steps are as follows:

1) Cleaning of images and artifacts;
2) Detection of the lesion segment;
3) Extracting required features
4) Classification.

Segmentation is an important step among them. Further processing is performed over the area where the segment lies on the image. If we don't have an efficient segment, the result can be quite misleading. There have been many approaches towards segmentation of Dermoscopic images.

G. Subha et al. [2] used Neural Network and related approaches towards the detection of cancerous lesion. The authors discuss various models like Radial basis Neural Network, Back Propagation based network which is generally a Multi-Layer Perceptron model and Extreme learning machine approach. A comparative performance survey was demonstrated. In [1], all the common and popular segmentation approaches have been discussed and have been applied towards the skin cancer problem. Techniques include Adaptive thresholding, Gradient Vector flow, Adaptive snakes, Level Set Methods, Expectation Maximization algorithm and fuzzy based approaches. Region growing and Region merging algorithm supported by Evolutionary model GA is discussed in [3]. In [4] an effective implementation of Neural Network based segmentation approach is discussed.

Neural Networks have always been debated in the literature and have been proven to generate better and promising results. In most cases, it acts like a good heuristic algorithm and in some sense we have limited information on how the algorithm is able to perform with an edge over traditional approaches. Also, Neural Networks are quite fast due to inherent parallelism. The reason for this is that each node in a Neural Network is essentially its own autonomous entity and each performs only a small computation in the grand-scheme of the problem and the aggregate of all these nodes, the entire network, is where the true capability lies [5].

Self-Organized Feature Maps (SOFM) are a sub domain of Neural Networks which is generally applied to clustering purposes. As the name suggests, SOFMs are unsupervised algorithms and they learn to organize their decisions in unison with other neurons towards the problem in focus. They have been used in many applications involving clustering, classification and speech/text recognition.

Self-Organized feature maps are directly related to the network model developed by Kohonen [9]. The Self-Organizing Map has the special property of effectively creating spatially organized "internal representations" of various features of input signals and their abstractions thereby enabling them to recognize semantics in various situations [9]. In a general SOFM system, only few neurons get the activation signal and due to the location of these neurons in

the map, the whole system tends to be ordered as if some meaningful coordinate system has been generated [9].

To our knowledge, there is little literature on using SOFMs towards the problem of Dermoscopic skin cancer detection and hence that is one of the main motivations in presenting this work.

2 Self-Organized Feature Maps: Details

2.1 SOFM in General

Technically, the SOFM learns from examples by mapping (projection) from a high-dimensional continuous input space onto a low-dimensional discrete space (lattice) of N neurons which are arranged in fixed topological forms, e.g., as a rectangular 2-dimensional array [7]. A rectangular field is preferable for easy computational and array based approaches.

In SOFM, the neurons learn by unsupervised competitive learning amongst the other neurons and they try to map their weights in accordance to the input [5]. We present our input to the neurons in a desired way and the neurons try to learn those inputs and in the end up mimicking the inputs roles. The resulting map preserves the topology of the input samples in the sense that adjacent patterns are mapped into adjacent regions on the map and due to this topology-preserving property, the SOFM is able to cluster input information and spatial relationships of the data on the map [7]. In this way, a novel input can be easily shown to the network and its right niche or group can be found out. Competitive learning is also called Winner take all Networks where from a whole field of neurons, only the winning neuron is able to learn positively.

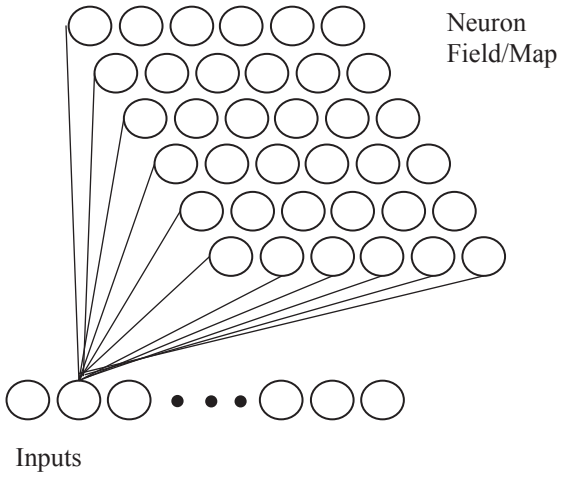

Figure 1. SOFM model [10]

The final organized Feature Map can be used for a lot of purposes out of which few are Clustering, Exploratory data analysis [6], visualization, removal of redundancy in data etc.

2.2 SOFM model

Here, we will see an example of a Neural Map and how the inputs are connected to the neurons.

In Figure 1, we can see a 6x6 map and a vector of inputs. The input vector is generated by the input values through some specific scheme. All the neurons are connected to all the input elements in the vector. Only one element and its connection is shown for better understanding. In this way the inputs are mapped to the neuron by vectors generated from the actual input values.

As the training process progresses, one of the neuron in the whole field wins and the weight gets mapped closer to its corresponding input. If we follow the winner take all rule, the winning neuron will get updated at the expense of others. This paper uses schemes where the update of other neurons involves dependency on their proximity to the winning neuron and that is seen in *section 2.3*

2.3 Mathematical Definitions

We can define the equation involved in the implementation of SOFM as follows:

As discussed earlier, that the SOFM field is generally in the shape of a square. For example in Figure 1, we see an example of a 6x6 field.

Let r_i be the i^{th} neuron in the SOFM field.

Let the number of inputs to each neuron be n.

Let x^k be the vector of n inputs and X be the set of input vectors.

Then we can define the weight matrix with respect to that neuron as:

$$w_i = [w_{i1}, w_{i2}, \ldots w_{in}]^T; \quad (1)$$

The training algorithm goes as follows:

1) Present $x^k \epsilon$ X(the set of input vectors)
2) Find the winning neuron as follows:
$$i^* = \text{argmin}||w_i - x_k|| \quad (2)$$

   After finding the winning neuron, we can determine the location in the field of neurons.

3) We can then update the other weight by using the schemes below:

$$\Delta w_{ij} = \eta \Lambda(i, i^*, t)(x_j^q - w_{ij}) \quad (3)$$

Where,

$$\Lambda(i, i^*, t) = \exp[\frac{-||r_i - r_{i*}||^2}{2\sigma^2(t)}] \quad (4)$$

&

$$\sigma(t) = \sigma_0 \exp(\frac{-t}{\tau_n}) \quad (5)$$

Where σ_0 and τ_n are fixed parameters. r_i is the location of the neuron in the field.

From the above equations we can see that as the training time increases, we tend to decrease to the neighborhood impact on the learning of the neurons. As the training progresses the different training data result into different wining neurons and the weights are updated according to it.

3 Segmentation Algorithms

This section deals with the different segmentation algorithms used for comparison including the proposed approach using Self Organizing Feature Maps.

3.1 Otsu's Method

3.1.1 Method in general.

Otsu's methods is a regularly utilized method in the problem of segmentation and it has been used against the skin cancer issue quite regularly. The basic assumption in the application of Otsu's method is the existence of a bimodal histogram or class of pixels [12]. Other assumptions include uniform illumination and less or no usage of spatial values i.e. the structure. It generally results into a binary image which is the output with the segment being highlighted and the rest becoming the background. The benefit of Otsu's method stems from the fact that it uses an iterative scheme to find that factor so that in the end the intra-class variance in the pixels is minimized or the inter-class variance is maximized.

3.1.2 Mathematical definitions.

Let us assume that that the grayscale values are within [0, L-1] which implies there are L different levels. Let's say that the algorithm divides the image at the grayscale level't'. It will then result into two classes of grayscale levels which are [0, t] and [t+1, L-1]. These two classes will then be tested for the within-class as well as between-class variances.

The class probabilities are given as follows:

$$Pr_1(t) = \sum_{i=0}^{t} P(i) \quad (6)$$

$$Pr_2(t) = \sum_{i=t+1}^{L-1} P(i) \quad (7)$$

The class means can be found as follows:

$$\mu_1(t) = \sum_{i=0}^{t} \frac{iP(i)}{Pr_1(t)} \quad (8)$$

$$\mu_2(t) = \sum_{i=t+1}^{L-1} \frac{iP(i)}{Pr_2(t)} \quad (9)$$

The weighted within class variance can be calculated as follows:

$$\sigma_w^2(t) = Pr_1(t)\sigma_1^2(t) + Pr_2(t)\sigma_2^2(t) \quad (10)$$

Where the individual class variances can be found as follows:

$$\sigma_1^2(t) = \sum_{i=0}^{t} [1 - \mu_1(t)]^2 \frac{P(i)}{Pr_1(t)} \quad (11)$$

$$\sigma_2^2(t) = \sum_{i=t+1}^{L-1} [1 - \mu_2(t)]^2 \frac{P(i)}{Pr_2(t)} \quad (12)$$

The most efficient't' can be found out and the segment will then be generated.

3.2 Fuzzy C Means

3.2.1 Method in general.

Fuzzy C Means is an algorithm which partitions a set of n objects such as $x=\{x_1, x_2, \ldots. x_N\}$ in R^d dimensional space to $C(1<C<N)$ fuzzy clusters with set $y=\{y_1, y_2, \ldots.y_C\}$ being the cluster heads/centroids of the fuzzy clusters [13].
The fuzzy association is defined by a matrix μ which is also called the fuzzy matrix. As one can notice, the dimensions of the matrix is NxC. For example μ_{ij}, an element in the i^{th} row and j^{th} column in the matrix represents the association of the i^{th} object with the j^{th} cluster. The process starts with random clusters and then the association of the objects is found based on Euclidian distance metric. The centroids are then recalculated using Equation (13). This process is repeated until the centroid allotments match successively.

3.2.2 Mathematical definitions.

FCM algorithm aims to minimize the following equation

$$J_m = \sum_{j=1}^{C} \sum_{i=1}^{N} \mu_{ij}^m d_{ij} \quad (13)$$

Where

$$d_{ij} = \|x_i - y_j\| \quad (14)$$

In the above equations, m defines the fuzziness. In our implementation, after many trials, we have selected m=2 as it

has shown better results. d_{ij} refers to the Euclidian distance between element x_i to cluster center in y_j.

3.3 Proposed approach using SOFM

Here, we describe our model and how we present the data to the neurons. Our data consists of skin cancer images as well as their segmented counterparts. Each image is of size 150x200 and its segment is also of the same size. We use data from an image and its segment together to form the training vector.

3.3.1 Neuron field.

Our neuron field consists of basically 150x200 neurons, each one of which caters to the decision at that pixel location as you can see from the size of the image. While training, we pass input vectors related to each pixel after consideration of both the images i.e. the image and its training segment; and this is done for all the pixels of an image. Also, similar steps are done for all the training images.
Since the size is 150*200, there are 30k neurons in our field and each neuron has weights of size 1x26. The reason of 26 weights is described in section 3.3.2.

3.3.2 Input vector generation.

The model which we are going to use is inspired by the one used in [10]. As discussed earlier, for every training image we have its corresponding segmented image. So let us denote the main image as X and its segmented image is Y.

Image X is in grayscale.
Image Y is in binary where '1' depicts segment pixel and '0' depicts surrounding pixel.

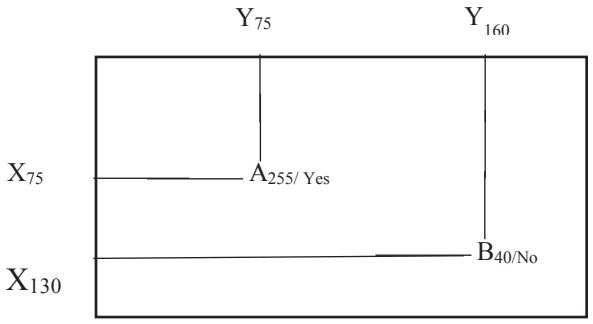

Figure 3. Image, pixel, location and grayscale value

While considering a certain pixel location at [a, b] where a and b are the coordinates;
• There will be a grayscale value (say g) in image X with 0<=g<=255.
• There will be binary value 0/1 in image Y.

We use the binary values of the coordinates, grayscale value in image X and the binary value in image Y altogether to form the input vector.
Mathematically, the input vector can be depicted as follows:

Inp_vec=[1x8 binary , 1x8 binary, 1x8 binary, 1x2 binary]
Where,The 'first' [1x8] binary is the binary value of the X-coordinate at the location of the pixel;

The 'second' [1x8] binary is the binary value of the Y-coordinate at the location of the pixel;
The 'third' [1x8] binary is the binary value of the grayscale value at the location of the pixel;
The 'fourth' [1x2] binary is the binary value of the segment decision which is sent as [1 0] for a segment pixel and [0 1] for a surrounding pixel.
So the total dimension of the Input Vector is [1x26].

For example in Figure 3, A and B are pixels. The subscript defines its grayscale value and whether it's a segment pixel or not. The coordinate locations are also mentioned.

Inp_Vec $_{(A)}$ = [01001011 01001011 11111111 10]
Inp_Vec $_{(B)}$ = [10000010 10100000 00101000 01]
The above vectors are sent for every pixel regarding one image and its segmented counterpart. Similar operations are done for all the training images.

3.3.3 Input vector generation while "Testing"

While testing, there is little change in the input vector. Only the last [1x2] binary matrix is replaced by [00]. We would only send the required location and pixel information. We let the network figure out whether the pixel should be a segment or not and we find it after thresholding the weights of the winning neuron.

4 Implementation and Comparisons

4.1 Datasets

In [17], Dermoscopy based work has been performed by the authors and the dataset including the results have been shared. It has been used as the training set mostly which consists of around 350 training images. Other testing source datasets are from dermis and dermquest which are public and thus open source.

4.2 Evaluation Methods

For the purpose of evaluation of our algorithm, we have selected five factors which are accuracy, sensitivity, specificity, Jaccard index and dice coefficient.
They can be calculated as follows:

$$Sensitivity = \frac{true_{positives}(TP)}{true_{positives}(TP)+false_{negatives}(FN)} \quad (15)$$

$$Specificity = \frac{true_{negatives}(TN)}{true_{negatives}(TN)+false_{positives}(FP)} \quad (16)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$

$$Jaccard_{index} = \frac{TP}{TP+FP+FN} \quad (18)$$

Jaccard Index is defined as the size of interaction of the two sets divided by the size of their union [14].

$$Dice_{coefficient} = \frac{TP}{\frac{1}{2}(TP+FN+TP+FP)} \quad (19)$$

Dice Coefficient is defined as the size of interaction of the two sets divided by their average size [15].

4.3 Table of measures

The values in table 1 are generated on average of all the *420* testing images together.

Segmentation Algorithm	Accuracy	Sensitivity	Specificity	Jaccard index	Dice coefficient
OTSU's Method	0.9464	0.7877	0.9643	0.7147	0.8179
FCM	0.9561	0.7960	0.9723	0.7080	0.8121
Proposed SOFM	0.9748	0.8624	0.9857	0.8586	0.9234

Table 1. Evaluation Parameters

4.4 Inferences

From Table 1 we can infer that, the proposed method has demonstrated better overall accuracy than the other two algorithms and this is an essential point. Sensitivity values are appreciable for the algorithms and are competitive amongst each other. The proposed method performs in a way which provides better results in comparison to Otsu's and FCM methods. It also shows improvements in other evaluation parameters.

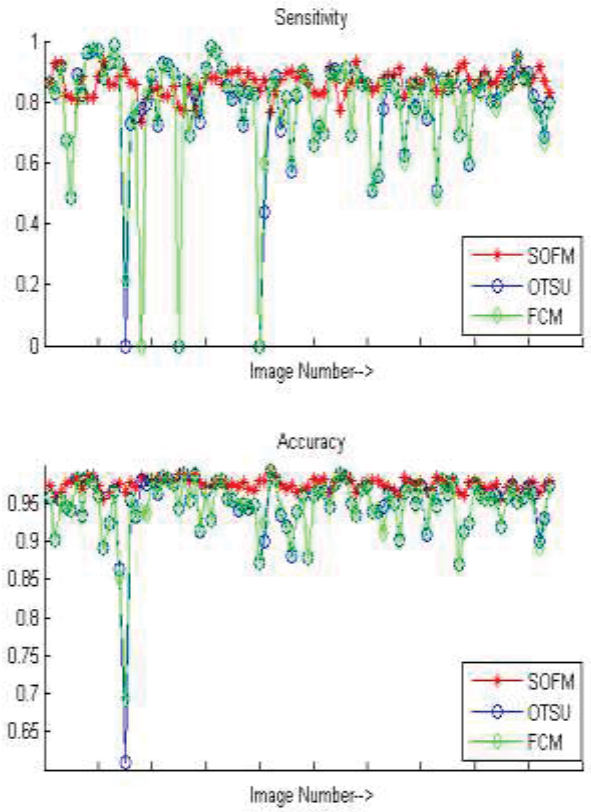

Figure 4: Comparison of Accuracy and Sensitivity of the algorithms

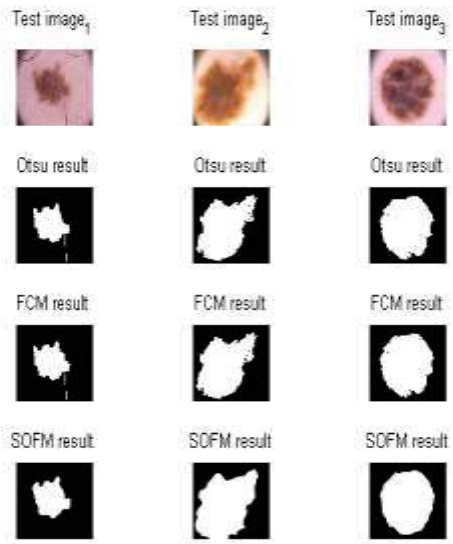

Figure 5: Comparison of Segmented images.

Figure 4 shows the variability of the parameters. We can see that the proposed method has comparatively better and consistent results which can be inferred by the lesser variability.

Figure 5 shows the different segments formed by the three algorithms and we can see that the proposed algorithm also works competitively against the other two.

5 Conclusion

As has been discussed in the paper, the proposed methodology has resulted relatively better results than the other existing methods. It has lesser fluctuation in the evaluation parameters which is a considerable sign of it being able to deal with novel data effectively. The main drawback noticed is the computational time in the proposed algorithm. However, after sufficient training, the method works very well in comparison to the other algorithms.

To further reinforce the findings of this paper, table 1shows how the methods have performed on an average over a large number of images together. We can see that our proposed algorithm has better performance when averaged over many test samples. Any discrepancies and variations in the results stem from the fact that all test samples are from entirely different datasets when compared to the training set. Further efforts will include improvements to the algorithm by adopting evolutionary/metaheuristic approaches.

6 References

[1] M Silveria, J C Nascimento, J S Marques, André R S Marcal, T Mendoca, S Yamauchi, J Maeda & J Rozeira, Comparison of Segmentation Methods for Melanoma diagonosis, in Dermoscopy images, IEEE journal of selected topics in Signal Processing, Vol 3 No 1, Feb-2009

[2] G S Vennila, L P Suresh & K L Shanmuganathan, Dermoscopic Image Segmentation and Classification using Machine Learning Algorithms, ICEET-2012

[3] Angelia S, L. Padma Suresh & S.H Krishna Veni, Image Segmentation based on Genetic Algorithm for region growth and region merging, ICEET-2012

[4] L Jinali & Z Baoqi, The Segmentation of skin cancer images based on genetic Neural Network.

[5] Shyam M. Guthikonda, Kohonen Self-Organizing Maps, Wittenberg University, December 2005

[6] A.Ultsch and H.P. Siemon, Kohonen's SeIf Organizing Feature Maps for Exploratory Data Analysis, Institute of Informatics University of Dortmund.

[7] Guilherme DE A. Barreto & Aluizio F. R. Araújo, Self-Organizing Feature Maps for modelling and control of robotic manipulators.

[8] R J Kuo, L M Ho & C M Hu, Integration of Self organizing feature map and K-means algorithm for market segmentation, Computers & Operations Research 29 (2002) 1475-1493, Elsevier Feb-2001.

[9] Teuvo Kohonen, The Self Organizing Map, Proceedings of the IEEE, VOL. 78, NO 9, SEPTEMBER 1990.

[10] Dmitri G Roussinov & Hsinchun Chen, A Scalable self-organizing map algorithm for textual classification: A Neural Network approach to thesaurus generation, Building the Interspace: Digital Library Infrastructure for a University Engineering Community," PIs: B. Schatz, H. Chen, et al., 1994-1998, IRI9411318

[11] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall, 1999

[12] Harpreet Kaur & Aashdeep Singh, Enhanced Skin Cancer detection techniques using Otsu segmentation method. IJARCSSE, Vol5 issue 5, May 2015

[13] Mahesh Yambal & Hitesh Gupta, Image Segmentation using Fuzzy C-means Clustering: A survey, IJARCCE, Vol. 2 Issue 7, July 2013.

[14] D A Santana-Calvo, Vargas Olivares, S Pichardo, L Curiel & JE Chong-Quero. Evaluation methods of image segmentation quality applied to magnetic resonance guided high-intensity focused ultrasound therapy. VI Latin American Congress on Biomedical Engineering CLAIB 2014, Paraná, Argentina 29, 30 & 31 October 2014, IFMBE Proceedings 49.

[15] David W Shattuck, Gautam Prasad, Mubeena Mirza, Katherine L. Narr, Arthur W Toga, Online resource for validation of brain segmentation methods, Elsevier-NeuroImage-Nov 25 2008.

[16] Aaron Fenster, Bernard Chiu, Evaluation of Segmentation algorithms, Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, September 1-4, 2005

[17] Teresa Mendonça,Pedro M. Ferreira, Jorge Marques, Andre R. S. Marcal, Jorge Rozeira, PH² - A Dermoscopic image database for research and benchmarking, 35th International Conference of the IEEE Engineering in Medicine and Biology Society, pp.5437-5440, July 3-7, 2013.