International conference on Advanced Techniques in Communication Networking and Automation

RESEARCH ARTICLE OPEN ACCESS

Human Nail Disease Diagnosis System Using DeepLearning

Arthi S

Priyadharshini J

Vidhya S

Nivetha R.M

B.E Computer Science and Engineering

A.V.C College of Engineering

Mannampandal, Mayiladuthurai

Devi Sri Nandhini.M

Assistant Professor

A.V.C College of Engineering

Mannampandal, Mayiladuthurai

nandhini.avcce@gmail.com

Abstract-Human's hand nail is analyzed to identify many diseases at early stage of diagnosis. Study of person hand nail color helps in identification of particular disease in healthcare domain. The proposed system guides in such scenario to take decision in disease diagnosis. The input to the proposed system is person nail image. The system will process an image of nail and extract features of nail which is used for disease diagnosis. Human nail consist of various features, out of which proposed system uses nail color changes for disease diagnosis. Here, first training set data is prepared from nail images of patients of specific diseases. A feature extracted from input nail image is compared with the training data set to get result. In this experiment we found that using color feature of nail image average 90% results are correctly matched with training set data during three tests conducted.

Keywords- Nail color, Input Image, Trained Data set,

I.Introduction

In healthcare domain many diseases can be predicted by observing color of human nails. Doctors observe nails of patient to get assistance in disease identification. Usually pink nails indicate healthy human. The need of system to analyze nails for disease prediction is because human eye is having subjectivity about colors, having limitation in resolution and small amount of color change in few pixels on nail would not be highlighted to human eyes which may lead to wrong result where as computer recognizes small color changes on nail.

II. Related Work

In reference [1] the nail diseases refer to some kind of deformity in the nail unit. Although the nail unit is a skin accessory, it has its own distinct class of diseases as these diseases have their own set of signs, symptoms, causes and effects that may or may not relate to other medical conditions. Recognizing nail diseases still remains an unexplored and a challenging endeavor in itself. This paper proposes a novel deep learning framework to detect and classify nail diseases from images. A distinct class of eleven diseases i.e. onychomycosis, subungulal hematoma, beau's lines, yellow nail syndrome, psoriasis, hyperpigmentation, koilonychias, paroncychia, pincer nails, leukonychia, and onychorrhexis. The framework uses a hybrid of Convolutional Neural Network (CNNs) for feature extraction. Due to the non-existence of a meticulous dataset, a new dataset was built for testing the enactment of our proposed framework. This work has been tested on our dataset and has also been compared with other state-of-the-art algorithms (SVM, ANN, KNN, and RF) that have been shown to have an excelled performance in the area of feature extraction. The results have shown a comparable performance, in terms of differentiating amongst the wide spectrum of nail diseases and are able to recognize them with an accuracy of 84.58%.

In reference [2] in this paper, a method and a device have been developed for the treatment of a resistant fungal infection with a complex of drugs and their delivery to the affected area of the nail plates. The developed method allows for effective treatment of fungal diseases of the nails of the hands and legs. The method is based on the use of such chemical compounds as silver nitrate, hydrogen peroxide.

In reference [3] Nails have a function or role that is very important to protect the soft fingertips and have a lot of nerves. In the medical world, several expert systems have begun to be used in helping doctors to diagnose a disease. This research was to detect abnormalities of the nails, Terry's nail. The textural characteristics are processed with grey level co-occurrence matrix (GLCM) and classifying method using KNN. The dataset in this study is taken from Google and also some of the paper that discusses the nail abnormalities. Nail pictures obtained are different from any source. Therefore, the image should be cut just one finger. Because when detecting terry's nail, the disorder usually occurs in all the nails. So we can use one finger. The photos of a nail that has been doing the extraction characteristics using GLCM then will be done using KNN classification. In this case the class will be divided into two classes, healthy and Terry's. From the experiment that have been done, the best accuracy results are 70.93% with 60-40 partition of dataset, K=1, light intensity values of 100-500 lux, a distance of 15 cm and an angle of 0^0

In reference [4] There are many types of nail diseases, and although the nail is just a small part of our body, the nail unit can be a significant sign of some underlying disease based upon its features. Subungual Melanoma remains a life-threatening disease. Although it can be cured in its early stages, it is difficult to diagnose it during that time. It often leads to a late disease diagnosis, which makes it difficult to cure the disease. The present medical tests for disease diagnosis are costly and not available in rural parts. This project proposes an AI approach to detect and classify nail diseases from images. A distinct class of two diseases i.e., yellow nail syndrome and Subungual Melanoma, is classified in this project. The project uses an Artificial Neural Network based model for training and testing. We have used the concept of transfer learning for the training model because making a model from scratch is not feasible with fewer data and less GPU. The model is an implementation of VGG16 by Keras framework with two added layers of ANN. Since we could not find any dataset, we made a new dataset for our proposed framework. This work has been tested on our dataset and has shown to have an excellent performance in identifying diseases.

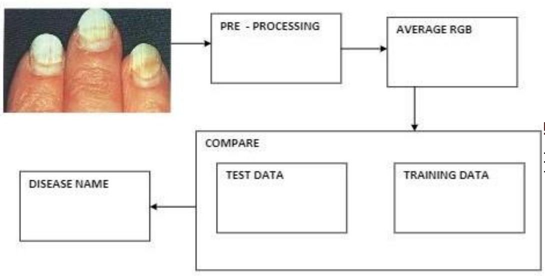
In reference [5] Human's hand nail is analyzed to identify many diseases at early stage of diagnosis. Study of person hand nail color helps in identification of particular disease in healthcare domain. The proposed system guides in such scenario to take decision in disease diagnosis. The input to the proposed system is person nail image. The system will process an image of nail and extract features of nail which is used for disease diagnosis. Human nail consist of various features, out of which proposed system uses nail color changes for disease diagnosis.

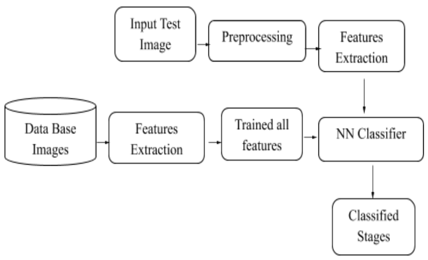
Here, first training set data is prepared using Weka tool from nail images of patients of specific diseases. A feature extracted from input nail image is comparedwith the training data set to get result. In this experiment we found that using color feature of nail image average 65% results are correctly matched withtraining set data during three tests conducted.

In reference [6] Human's hand nail is analysed to identify many diseases at early stage of diagnosis. Study of person hand nail colour helps in identification of particular disease in healthcare domain. The proposed system guides in such scenario to take decision in disease diagnosis. The input to the proposed system is person nail image. The system will process an image of nail and extract features of nail which is used for disease diagnosis. Human nail consist of various features, out of which proposed system uses nail colour changes for disease diagnosis. Here, first training set data is prepared using Weka tool from nail images of patients of specific diseases. A feature extracted from input nail image is comparedwith the training dataset to get result. In this experiment we found that using color feature of nail image average 65% results are correctly matched with training set data during three tests conducted. Finally, the early-stage diseases are diagnosed using the Human Nail.

In reference [7] Nails are specialized skin tissue. It can reflect some health conditions by wrinkles and texture on the surface. The physicians mostly diagnosed through observing first, and confirmed with biopsy examination, which was time consuming in clinical practice. Therefore, an instancesegmentation algorithm Mask R-CNN deep learning model was utilized in this research to segment and classify 17 kinds of potential lesions within the nail images. In the results, the Mask R-CNN model couldreach a precision rate of 87.69% in classification. For further comparison with the state-of-the-art DenseNet201, whose precision rate was 84.37%, the Mask R-CNN model could outperform CNN-based algorithms. Hopefully, with the strength of the Mask R-CNN algorithm, it will access great diagnosis of naildiseases in clinical practice

III. Proposed methodlogy





The proposed processes the image of the affected nail by the available preprocessing techniques. Input image is preprocessed to reduce the noises and for enhancing the input image for processing. After the stage of preprocessing averaging of RGB values are done to save the space and to reduce the size for making the processing easier. Then the input image is sent to the trained model where it is get compared with predefined available features of affected nail. Based on the analysis of input image over the trained model the respective disease can be identified for that particular disease.

PREPROCESSING:

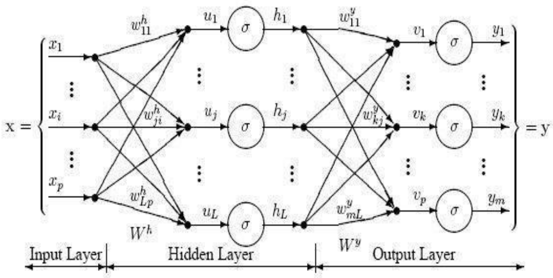
The aim of pre-processing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing. Image Pre-processing is a common name for operations with images at the lowest level of abstraction. Its input and output are intensity images. Image restoration is the operation of taking a corrupted/noisy image and estimating the clean original image. Corruption may come in many forms such as motion blur, noise, and camera misfocus. Image restoration is different from image enhancement in that the latter is designed to emphasize features of the image that make the image more pleasing to the observer, but not necessarily to produce realistic data from a scientific point of view. Image enhancement techniques (like contrast stretching or de-blurring by a nearest neighbour procedure) provided by "Imaging packages" use no a priori model of the process that created the image. With image enhancement noise can be effectively be removed by sacrificing some resolution, but this is not acceptable in many applications. In a Fluorescence Microscope resolution in the z-direction is bad as it is. More advanced imageprocessing techniques must be applied to recover the object. De-Convolution is an example of image restoration method. It is capable of: Increasing resolution, especially in the axial direction removing noise increasing contrast.

NEURAL NETWORK AND IT USED:

Neural network is the best tool in recognition and discrimination between different sets of signals. To get best results using the neural network, it is necessary to choose a suitable architecture and learning algorithm. Unfortunately there is no guaranteed method to do that. The best way to do that is to choose what is expected to be suitable according to our previous experience and then to expand or shrink the neural network size until a reasonable output is obtained. In this work we tried different sizes for the neural network using MAT LAB and we found that the best in our case is the model shown in Fig. 2. It has an input layer with 2000 inputs, first hidden layer with 11 nodes, and T ANSIG transfer function, second hidden layer with 7 nodes, and T ANSIG transfer function, and output layer with PURELIN transfer function and 2 outputs. One of the two outputs is used for the detection of tumor, and the other for the localization. T ANSIG transfer function is selected to limit the signal between -1 and 1. For the output layer, PURELIN transfer function is chosen to give all the possible cases for the location of tumor.

The proposed neural network has been used to detect and locate the tumor in two cases. The first case was to detect and locate a tumor in a two-dimensional sector of the cervical model. The location of tumor was considered randomly at the centerand in any of the four quadrature. The second case was to detect and locate tumor anywhere in the three-dimensional model. The neural network has been trained using 100 sets of inputs using the training function (TRAINSCG). Additional 40 Sets of inputs were used to test the performance of each neural network.

The Multilayer Perceptron Neural Network Model



This network has an **input layer** (on the left) with three neurons, one **hidden layer** (in the middle) with three neurons and an **output layer** (on the right) with three neurons. There is one neuron in the input layer for each predictor variable. In the case of categorical variables, *N*-1 neurons are used to represent the *N* categories of the variable.

Input Layer — A vector of predictor variable values ($x_1 \ldots x_p$) is presented to the input layer. The input layer (or processing before the input layer) standardizes these values so that the range of each variable is -1 to 1. The **input layer** distributes the values to each of the neurons in the hidden layer. In addition to the predictor variables, there is a constant input of 1.0, called the *bias* that is fed to each of the hidden layers; the bias is multiplied by a weight and added to the sum going into the neuron.

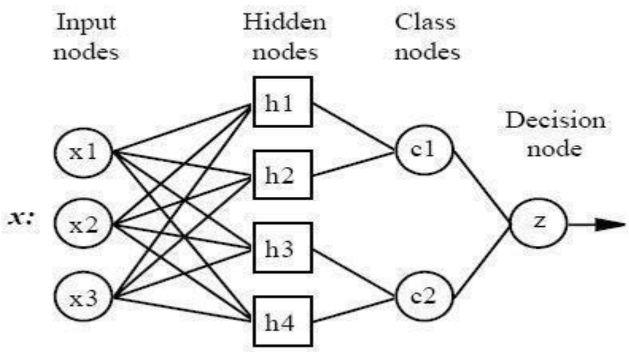
2. The **Hidden Layer** Arriving at a neuron in the hidden layer, the value from each input neuron is multiplied by a weight (w_{ji}), and the resulting weighted values are added together producing a combined value u_j. The weighted sum (u_j) is fed into a transfer function, σ, which outputs a value h_j. The outputs from the hidden layer are distributed to the output layer.

3. The **Output Layer** Arriving at a neuron in the output layer, the value from each hidden layer neuron is multiplied by a weight (w_{kj}), and the resulting weighted values are added together producing a combined value v_j. The weighted sum (v_j) is fed into a transfer function, σ, which outputs a value y_k. The *y* values are the outputs of the network.

Neural Networks (NN) and General Regression Neural Networks (GRNN):

Neural Networks (NN) and General Regression Neural Networks (GRNN) have similar architectures, but there is a fundamental difference: networks perform classification where the target variable is categorical, whereas general regression neural networks perform regression where the target variable is continuous. If you select a NN/GRNN network, DTREG will automatically select the correct type of network based on the type of target variable.

Architecture of a (NN) and (GRNN):



All NN networks have four layers:

1. **Input layer** — There is one neuron in the input layer for each predictor variable. In the case of categorical variables, *N*-1 neurons are used where *N* is the number of categories. The input neurons (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

1. **Hidden layer** — This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along

with the target value. When presented with the *x* vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function using the sigma value(s). The resulting value is passed to the neurons in the pattern layer.

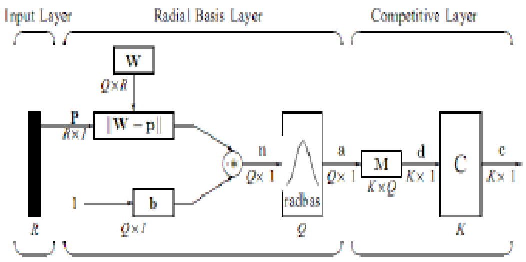
1. **Pattern layer / Summation layer** — The next layer in the network is different for NN networks and for GRNN networks. For NN networks there is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron's category. The pattern neurons add the values for the class they represent (hence, it is a weighted vote for that category).

For GRNN networks, there are only two neurons in the pattern layer. One neuron is the denominator summation unit the other is the numerator summation unit. The denominator summation unit adds up the weight values coming from each of the hidden neurons. The numerator summation unit adds up the weight values multiplied by the actual target value for each hidden neuron.

1. **Decision layer** — The decision layer is different for NN and GRNN networks. For NN networks, the decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

For GRNN networks, the decision layer divides the value accumulated in the numerator summation unit by the value in the denominator summation unit and uses the result as the predicted target value.

The following diagram is actual diagram or propose network used in our project :



1) Input Layer:

The input vector, denoted as **p**, is presented as the black vertical bar.Its dimension is *R* × *1*. In this paper, *R* = *3*.

2) Radial Basis Layer:

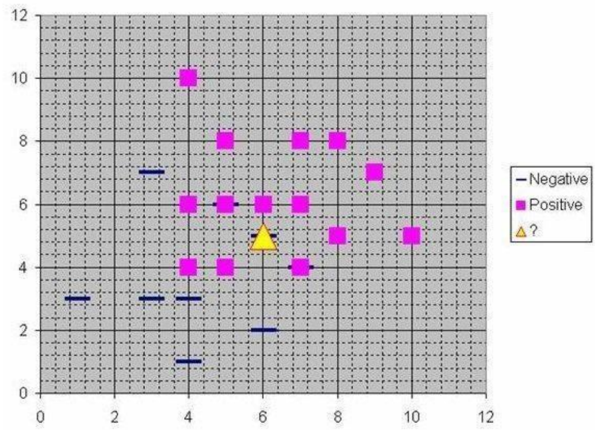
In Radial Basis Layer, the vector distances between input vector p and the weight vector made of each row of weight matrix **W** are calculated. Here, the vector distance is defined as the dot product between two vectors [8]. Assume the dimension of **W** is *Q*×*R*. The dot product between **p** and the *i*-th row of **W** produces the *i*-th element of the distance vector ||**W**- **p**||, whose dimension is *Q*×*1*. The minus symbol, "**-**", indicates that it is the distance between vectors. Then, the bias vector **b** is combined with ||**W**- **p**|| by an element-by-element multiplication, .The result is denoted as **n** = ||**W**- **p**|| ..**p**. The transfer function in NN has built into a distance criterion with respect to a center. In this paper, it is defined as *radbas(n)*= 2 *n e*- (1) Each element of **n** is substituted into Eq. 1 and produces corresponding element of **a**, the output vector of Radial Basis Layer. The *i*-th element of **a** can be represented as ai = *radbas*(||**W**i - **p**|| ..**b**i) (2) where **W**i is the vector made of the *i*-th row of **W** and **b**i is the *i*-th element of bias vector **b**.

3) Competitive Layer:

There is no bias in Competitive Layer. In Competitive Layer, the vector **a** is firstly multiplied with layer weight matrix **M**, producing an output vector **d**. The competitive function, denoted as **C** in Fig. 2, produces a 1 corresponding to the largest element of **d**, and 0's elsewhere. The output vector of competitive function is denoted as **c**. The index of 1 in **c** is the number of tumor that the system can classify. The dimension of output vector, *K*, is 5 in this paper.

How NN network work:

Although the implementation is very different, neural networks are conceptually similar to *K-Nearest Neighbor* (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure



Assume that each case in the training set has two predictor variables, *x* and *y*. The cases are plotted using their *x*,*y* coordinates as shown in the figure. Also assume that the target variable has two categories, *positive* which is denoted by a square and *negative* which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values *x*=6, *y*=5.1. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

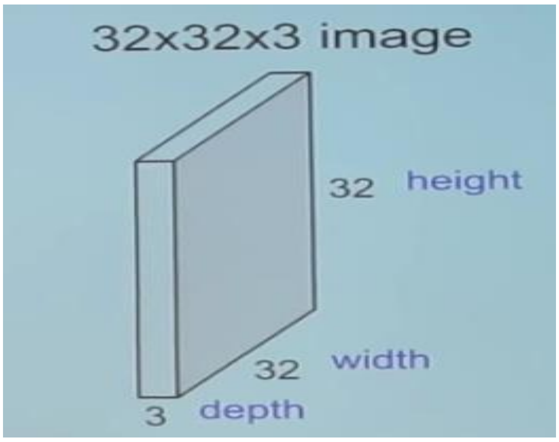
The nearest neighbor classification performed for this example depends on how many neighboring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

A neural network builds on this foundation and generalizes it to consider all of the other points. The distance is computed from the point being evaluated to each of the other points, and a *radial basis function* (RBF) (also called a *kernel function*) is applied to the distance to compute the weight (influence) for each point. The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (*distance*)

The further some other point is from the new point, the less influence it has.

Example of a RGB image (let's call it 'input image')

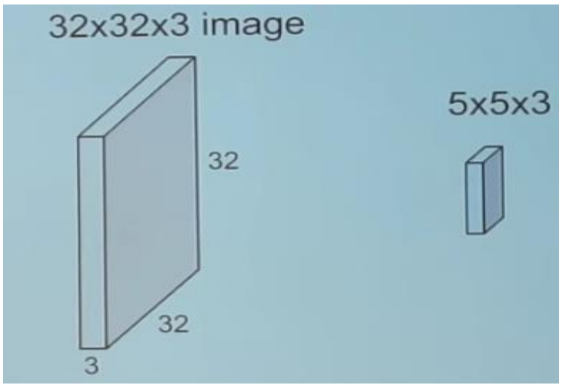


Unlike neural networks, where the input is a vector, here the input is a multi-channeled image (3 channeled in this case).

There are other differences that we will talk aboutin a while

Before we go any deeper, let us first understand what convolution means.

Convolution



Convolving an image with a filter

We take the 5*5*3 filter and slide it over the complete image and along the way take the dot product between the filter and chunks of the input image.

This is how it looks

For every dot product taken, the result is a scalar.

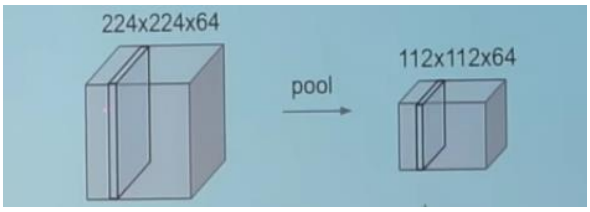
So, what happens when we convolve the complete image with the filter?

Convolution Layer

The convolution layer comprises of a set of independent filters (6 in the example shown). Each filter is independently convolved with the image and we end up with 6 feature maps of shape 28*28*1.Suppose we have a number of convolution layers in sequence.

Pooling Layers

A pooling layer is another building block of a CNN.



Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently.

IV. Results and Discussion

In the proposed technique we have trained a model that classifies the disease based on the pattern on the nail. This proposed system is able to predict the disease for the respective pattern of the nail with high accuracy. It is able to identify the small patterns also such that providing a system with higher success rate. The limitations of the existing model are eliminated by the proposed model.

References

1. M. H. Memon, J. Li, A. U. Haq and M. Hunain Memon, "Early Stage Alzheimer's Disease Diagnosis Method," *2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing*, Chengdu, China, 2019, pp. 222-225, doi: 10.1109/ICCWAMTIP47768.2019.9067689.

2. R. Nijhawan, R. Verma, Ayushi, S. Bhushan, R. Dua and A. Mittal, "An Integrated Deep Learning Framework Approach for Nail Disease Identification," 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Jaipur, 2017, pp. 197-202, doi: 10.1109/SITIS.2017.42.

3. H. Pandit and D. M. Shah, "A system for nail color analysis in healthcare," 2013 International Conference on Intelligent Systems and Signal Processing (ISSP), Vallabh Vidyanagar, India, 2013, pp. 221-223, doi: 10.1109/ISSP.2013.6526906.

4. L. Safira, B. Irawan and C. Setianingsih, "K- Nearest Neighbour Classification and Feature Extraction GLCM for Identification of Terry's Nail," 2019 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), BALI, Indonesia, 2019, pp. 98- 104, doi: 10.1109/ICIAICT.2019.8784856.

5. J. Gao, "Computer-Vision-Based Identification of Nail's Surface Shape," 2009 International Conference on Computational Intelligence and Security, Beijing, China, 2009, pp. 344-347, doi: 10.1109/CIS.2009.179.

6 .R. Nijhawan, R. Verma, Ayushi, S. Bhushan, R. Dua and A. Mittal, "An Integrated Deep Learning Framework Approach for Nail Disease Identification,"

2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), 2017, pp. 197-202, doi: 10.1109/SITIS.2017.42.

7. E. Olenev, W. Al-Haidri and O. Lebedinskaya, "Method and Device for Treatment of Fungal Diseases of the Skin and Nails," 2019 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), 2019, pp. 25-27, doi: 10.1109/USBEREIT.2019.8736581.

8. L. Safira, B. Irawan and C. Setianingsih, "K-Nearest Neighbour Classification and Feature Extraction GLCM for Identification of Terry's Nail," 2019 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), 2019, pp. 98-104, doi: 10.1109/ICIAICT.2019.8784856.

9. M. Mehra, S. D'Costa, R. D'Mello, J. George and D. R. Kalbande, "Leveraging Deep Learning for Nail Disease Diagnostic," 2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE), 2021, pp. 1-5, doi: 10.1109/ICNTE51185.2021.9487709.

10. R. Sharine, J. Sharon and G. S. Uthayakumar, "Early Stage Disease Diagnosis Using Human Nail In Image Processing", *International Journal Of Advanced Research In Science Communication and Technology (IJARSCT)*, vol. 4, no. 1, April 2021, ISSN ISSN 2581-9429.

11. L. Alzubaidi, M. Al-Amidie, A. Al-Asadi, A. J. Humaidi, O. Al-Shamma, M. A. Fadhel, J. Zhang, J. Santamar´ıa and Y. Duan, Novel transfer learning approach for medical imaging with limited labeled data, Cancers (Basel), vol.13, no.7, DOI: 10.3390/cancers13071590, 2021

12. Sharma Vipra and Manoj Kumar Ramaiya, "Nail Color and Texture Analysis for Disease Detection", *BSBT 2015*, 2015.

13. U Wollina, P Nenoff, G Haroske and HA Haenssle, "The Diagnosis and Treatment of Nail Disorders", *Dtsch Arztebl Int.*, vol. 113, no. 29–30, pp. 509-518, 2016.

14. Mente Rajivkumar, "A Review: Fingernail Images for Disease Detection", *International Journal of Engineering and Computer Science*, vol. 6, 2017.