International Journal of Communication Networks and Information Security
2024, 16(1), 6251
ISSN: 2073-607X,2076-0930
https://ijcnis.org/

Research Article

Deep Belief Neural Network Framework for an Effective Scalp Detection System Through Optimization

Vijitha Khan [1*], Kamalraj Subramaniam [2]

[1] Research Scholar, Faculty of Engineering, Department of Electronics and Communication Engineering, Karpagam Academy of Higher Education, Coimbatore, India
[2] Professor and Head, Faculty of Engineering, Department of Biomedical Engineering, Karpagam Academy of Higher Education, Coimbatore, India
***Corresponding Author:** vijithakhan21@gmail.com

Citation: V. Khan and D. K. Subramaniam "Deep Belief Neural Network Framework for an Effective Scalp Detection System Through Optimisation," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 16, no. 1, pp. 46-67, Jan. 2024.

ARTICLE INFO

Received: 14 Dec 2023
Accepted: 20 Mar 2024

ABSTRACT

In an era where technology rapidly enhances various sectors, medical services have greatly benefited, particularly in tackling the prevalent issue of hair loss, which affects individuals' self-esteem and social interactions. Acknowledging the need for advanced hair and scalp care, this paper introduces a cost-effective, tech-driven solution for diagnosing scalp conditions. Utilizing the power of deep learning, we present the Grey Wolf-based Enhanced Deep Belief Neural (GW-EDBN) method, a novel approach trained on a vast array of internet-derived scalp images. This technique focuses on accurately identifying key symptoms like dandruff, oily scalp, folliculitis, and hair loss. Through initial data cleansing with Adaptive Gradient Filtering (AGF) and subsequent feature extraction methods, the GW-EDBN isolates critical indicators of scalp health. By incorporating these features into its Enhanced Deep Belief Network (EDBN) and applying Grey Wolf Optimization (GWO), the system achieves unprecedented precision in diagnosing scalp ailments. This model not only surpasses existing alternatives in accuracy but also offers a more affordable option for individuals seeking hair and scalp analysis, backed by experimental validation across several performance metrics including precision, recall, and execution time. This advancement signifies a leap forward in accessible, high-accuracy medical diagnostics for hair and scalp health, potentially revolutionizing personal care practices.

Keywords: Gradient Filtering; Techniques for Centroid and Invariant Moment; Improved Neural Belief Neural; Scalp Detection and Optimisation of Grey Wolves.

INTRODUCTION

Hair significantly influences a person's appearance. It's made up of a thin outer layer that repels water, accounting for about 1-8% of its structure. The bulk of hair, around 80-90%, is keratin, a tough protein that shapes into spirals or sheets. Along with these, hair has a bit of melanin, less than 3%, which determines its color [1]. This mix not only gives hair its unique look but also its strength. External factors like climate changes, humidity, temperature fluctuations, and exposure to chemical or physical treatments affect the keratin layer of hair, causing it to become brittle and develop cracks, thereby impacting its quality [2]. Yet, comprehensively studying hair remains challenging due to its intricacies, resulting in limited research on hair damage despite its susceptibility to various issues [3, 4]. Hair is susceptible to damage caused by various chemical compounds and physical elements like combing, dyeing, detergents, and exposure to ultraviolet radiation [5]. The widespread utilization of Scanning Electron Microscopy (SEM), particularly in microscopy advancements, has enabled the detailed examination of microscopic hair characteristics.

However, a comprehensive quantitative classification of hair damage typically relies on manual techniques

Copyright © 2024 by Author/s and Licensed by IJCNIS. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

such as Energy- SEM photographs of microscopic hair structures are processed using a variety of physical and chemical techniques, including dispersive X-ray (EDX) spectroscopy [6]. Many people suffer from scalp and hair problems, including dandruff, folliculitis, hair loss, and oily hair. These conditions are frequently linked to bad daily routines, unbalanced diets, elevated stress levels, and environmental contaminants. These scalp issues are the focus of recent advancements in specialized treatments like nontraditional hair therapy [7]. Nearly 70% of adults experience scalp hair disorders, according to the World Health Organization (WHO), which may be related to internal variables such as hormone imbalances, genetic predispositions, medical illnesses, or other underlying issues [8].

The evaluation of scalp hair quality within existing standard procedures for scalp hair treatment is predominantly done manually. Concerns about hair loss are prevalent among women, men, and children [9]. Hair loss often signifies an underlying medical issue—some stemming from genetics, while others are temporary, triggered by illness, stress, cancer treatments, weight loss, or iron deficiency. The impact of baldness or hair loss extends beyond mere physicality, significantly affecting both social life and overall health [10]. Consequently, there's been a heightened emphasis on the serious consideration of hair and scalp care.

Although professional hair salons or medical and cosmetic clinics offer hair care and scalp assessments, these services come at a significant cost [11]. However, advancements in the computing power of smart devices have led to the availability of more affordable hair and scalp analysis systems. Hair loss can result from various factors such as ageing, gender, illnesses, hereditary traits, stress, misuse of hair care products, inadequate nutrition, and exposure to diverse temperature zones. Notably, hair loss has increasingly affected younger individuals in recent times [12]. The psychological effects, such as lowered self-esteem brought on by thinning hair and an early receding hairline, encourage people to pay more attention to hair and scalp health, with many turning to specialist hair care shops or medical and cosmetic clinics for assistance [13].

Delays in treating hair loss frequently lead to eventual baldness or negative health implications, regardless of how expensive these treatments may be. Being a skin component, the scalp may lose hair as a result of external variables or underlying medical issues in an individual [14]. Many disorders of the scalp, including folliculitis, seborrheic dermatitis, psoriasis, allergies to the scalp, and dry or oily scalp, add to the increasing need for reasonably priced instruments that allow for quick diagnosis of scalp problems [15, 16].

In order to provide a comprehensive assessment of scalp images, numerous research projects have examined various aspects of scalp conditions, including health state, diameter, density, oiliness, and the number of individual hair follicles. With technological advancements, there's been a growing interest in preventing and remedying hair damage. However, traditional observational methods face challenges in swiftly and precisely identifying areas of hair breakage. Recent strides in artificial intelligence have presented opportunities for researchers by introducing advanced deep-learning models tailored for detecting hair and scalp damage. This study introduces an effective model that leverages deep learning techniques to automatically detect and categorize damage in hair surface images, utilizing SEM images and scalp photographs.

Leveraging grey wolf optimization with a deep belief neural network enhances the precision of hair damage detection. The primary objective of this designed model is to accurately identify and diagnose scalp issues within a reduced execution time. Furthermore, there's a focus on refining feature extraction techniques through centroid and invariant moment methodologies.

Below is a summary of the designed model's primary contribution;

Initially, the system is taught using separate collections of scalp hair and SEM pictures.

Next, create the Grey Wolf-based Enhanced Deep Belief Neural (GW-EDBN) model to precisely forecast issues with the hair and scalp.

A pre-processing method called Adaptive Gradient Filtering (AGF) is employed to remove noise from the dataset [17].

Using centroid and invariant moment approaches, feature extraction is carried out to extract the pertinent features [18].

Following that, in order to categorise scalp detection, the retrieved features are updated to the classification layer using EDBN [19].

Grey wolf optimization (GWO) [20] is used to optimize the EDBN parameters in order to accurately detect symptoms related to scalp hair.

Lastly, the outcomes of the developed model are compared to other existing models in terms of precision, recall, precision, time to execution, error rate, and F-measure.

LITERATURE REVIEW

Analysis of Energy-Carrying Collaborative Communication Technology

A Convolutional Neural Network (CNN) with a Deep Learning (DL) foundation was presented by Lintong et al. [21] to effectively detect hair damage from scanning electron microscope (SEM) images. Their model distinguishes variations in the hair surface indicative of diverse levels of damage and employs CNN for this detection. While successfully categorizing damage into weak, moderate, and high levels, their model encounters challenges associated with over fitting.

In an independent investigation, Ming-Che Chen et al. [22] introduced Scalp Eye, a scalp diagnostic system integrating Deep Learning (DL) and Artificial Intelligence (AI) methodologies aimed at improving scalp health monitoring. Scalp Eye integrates a portable microscope, captures scalp images, utilizes cloud infrastructure, incorporates a mobile application, and relies on a training server to identify scalp hair symptoms. While achieving a precision rate of 98.25% in comparison to existing models, Scalp Eye faces operational challenges such as vanishing gradient issues and errors during execution.

The author of [23] developed an expert system with the goal of precisely diagnosing diverse hair loss conditions including ringworm, lichen planus, seborrhea, and baldness. Utilizing a dataset comprising images of hair loss and expert language snippets, this method assists physicians in making diagnoses. Although validation with medical students demonstrated improved performance, there was a clear need for enhanced accuracy.

Furthermore, the Norwood-Hamilton technique for determining and evaluating scalp surface state was introduced by the author of [24]. Utilizing a microscope and webcam camera sensors, this method extracts image characteristics crucial for evaluating scalp and hair conditions, offering users vital information about their physical state. Although enhancing computational efficiency, this approach encounters challenges with classification.

The study of [25] looked into the use of corresponding photos to classify hairy scalp issues using machine learning (ML) and deep learning (DL) models. They used classification learner apps to compare and apply different DL and ML algorithms, and they were able to categorise these photos with an accuracy of 89.77%. However, this model experiences increased execution time in classifying hairy scalp issues.

A multifunctional Mask R-CNN framework was developed by the author of [26] to categorise hair follicles and evaluate the severity of hair loss. Initially, ResNet systems were trained on small scalp images, expanded, enhanced, and then utilized for feature extraction analysis. This model evaluated hair loss degrees in specific scalp areas based on individual follicle conditions, enhancing classification accuracy by 4 to 15%. However, it encounters a challenge with a low recall rate. Hansoo, et al [27], introduced a groundbreaking deep-learning model capable of swiftly and accurately identifying various scalp conditions without needing additional scalp information. This method boasts high precision, achieving an average accuracy of 98.78%. Nevertheless, it faces issues related to vanishing gradients. Jeong, et al [28], proposed the Scalp Grader model—an Artificial Intelligence (AI)-based mechanism for diagnosing and categorizing scalp disorders. By retraining system images using EfficientNet and CNN learning models, this approach identifies and classifies scalp conditions with accuracies ranging from 87.3% to 91.3%. However, it contends with a high error rate. Minjeong et al [29] developed a highly accurate model specialized in alopecia analysis. Utilizing a collection of alopecia photos, including varying severity levels, normalization, and augmentation techniques significantly improved the dataset. This DL model received an accuracy rating of 95.75% and an F1 score of 87.05% when compared to other ensemble models. Nevertheless, this model necessitates increased execution time.

Seung Hyun [30] introduced an inventive method for categorizing and scoring scalp issues—the Scalp Photographic Index (SPI). This system rates five characteristics of scalp disorders on a scale of 0 to 3 using a trichoscope. The correlation between dermatologist scalp evaluations and SPI grading proved strong for each of the five scalp aspects. While SPI grading demonstrated significant internal consistency and acceptable reliability, it faced challenges with classification accuracy. On another front, Mrinmoy Roy [31] devised a deep-learning approach to predict hair loss and scalp-related conditions. After gathering 150 images from various sources, the information was cleaned up and added to a 2D CNN model. The model's validation rate was 91.1%, while its training accuracy was 96.2%. However, issues emerged concerning over fitting and vanishing gradients (Table 1).

Table 1. Below is a full synopsis of the Literature Survey

Reference	Technique	Advantages	Disadvantages
[21]	DL based CNN	Better classification results Improve efficiency	Over fitting problem
[22]	Scalp Eye	Better detection High precision	Vanishing gradient problem Error
[23]	Effective Expert System	Improve hair scalp diagnosis system Less error rate	Less accuracy
[24]	Norwood-Hamilton Technique	Improve computing efficiency Provide actual information	Classification problem
[25]	Experimental Evaluation of ML and DL Models	Classify the hairy scalp problem	High execution time Less accuracy
[26]	Mask R-CNN	A more precise and efficient algorithm Increasing classification accuracy	The recall rate is low. Error
[27]	Surface-Sensing CNN	Gained better accuracy of 98.78% Less error rate	Vanishing gradient issue
[28]	AI-Scalp Grader	Data reliability ranged from 87.3 to 91.3% Improve dataset	The error rate is high. High cost
[29]	DL Based Scalp Image Analysis	Better accuracy and F1 score High effectiveness	The model required more execution time
[30], [31]	Development of a New Classification System	SPI grading revealed good correlations Great internal consistency and acceptable reliability.	Less reliability Classification accuracy is low.
[31]	Hair and Scalp Disease Detection System	Better training and testing accuracy High scalability	Over fitting and disappearing gradient issues

Numerous research endeavours have been dedicated to detecting hair and scalp issues, employing various methodologies such as CNN, expert systems, Scalp Eye, ML, and the Norwood-Hamilton model. However, despite these efforts, finding an accurate and fitting solution has been challenging due to several persistent issues. The challenges include low classification and detection precision, vanishing gradient problems, over fitting, long running times, errors, complex data, difficulties in identifying and forecasting scalp conditions, problems with classification, and persistent over fitting issues.

An optimization-focused Enhanced Deep Belief Neural (EDBN) model was developed to address these important issues. The goal of this model is to generate precise outcomes within the hair scalp system while overcoming optimisation obstacles. Furthermore, it significantly enhances the efficacy of scalp and hair diagnostic devices, leading to better experimental results.

METHODOLOGY

In this study, we developed a special system called GW-EDBN, inspired by Grey Wolf behavior, to spot and figure out hair and scalp issues by looking at photos of scalps found online. This system is really good at noticing four main problems: oily hair, folliculitis (which is like scalp pimples), dandruff, and hair loss. To make sure we're really accurate, we first clean up these photos using a method called Adaptive Gradient Filtering (AGF) to get rid of any visual mess or mistakes. Then, we use special techniques to pick out important details from these images, which help us understand what's going on with the scalp. After finding these key details, we feed them into our system to help it recognize scalp issues. We also make our system smarter using a strategy called grey wolf optimization (GWO), fine-tuning it to catch scalp problems more accurately. Thanks to all these steps, our

system gets really good at telling apart different hair conditions and spotting scalp problems, making it a great tool for understanding hair health better.

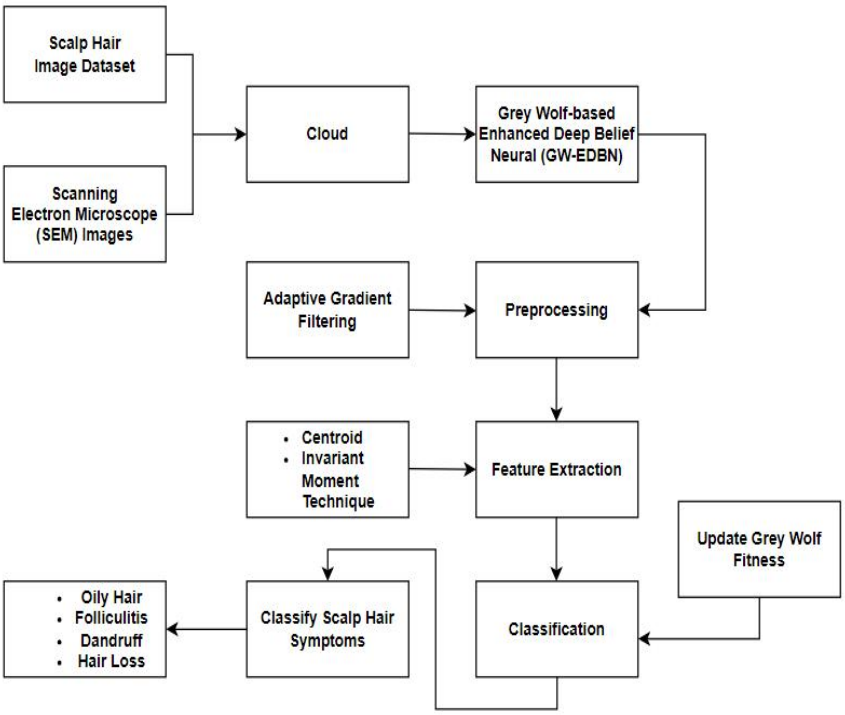


Figure 1. Suggested Approach

In this study, we used several tools written in Python, like TensorFlow, Scikit-learn, Pandas, NumPy, and Matplotlib, to run our experiments. TensorFlow helped us build and train complex models for deep learning, such as the Deep Belief Neural Network (DBNN), and we made it even better with something called grey wolf optimization. Scikit-learn gave us a bunch of useful methods for organizing our data, picking out important bits, and figuring out how well our models worked. Pandas and NumPy were key for dealing with our data and doing math stuff quickly and accurately, like getting our data ready, managing lists of features, and all sorts of calculations. Lastly, we used Matplotlib to make charts that show how our experiments went, comparing how different models did over time.

We picked these Python tools because they're well-liked, easy to use, have all the features we need for our study, and work well with other tools we're using. They also are known for working fast, being able to handle big projects, and having a lot of people using them who can help out. This makes sure we can do our experiments without too many hitches and that others can repeat what we did if they want to.

Pre-Processing Using Adaptive Gradient Filtering (AGF)

When working with tasks like sorting things into groups or predicting values, Adaptive Gradient Filtering (AGF) is a super useful step to prepare your data. It's especially good for dealing with data that has lots of details, is pretty noisy, or has complex relationships that aren't straight lines. AGF helps clean up images by getting rid of the unwanted noise but keeps the important patterns untouched. This makes everything clearer and easier to work with.This process involves the identification of noise by contrasting each pixel with its neighbouring pixels, allowing the adaptive filter to distinguish and label noise accordingly.

The dataset includes greyscale, which changes more in noise than background areas. Calculate the sliding window's grayscale variation S(a,b) as it scans the image to evaluate the noise characteristics of each pixel. Next, possible noise-candidate pixels undergo the filtering process. Suppose that j shows all eight directions and that the gradient in each direction is $\nabla J_j(a,b)$. The following is the definition of the judgment rule found in Equations (1) and (2):

$$\nabla J_j(a,b) = \frac{\sum_{j=0}^{7} \nabla J_j(a,b)}{8} \qquad (1)$$

$$N\nabla J(a,b) \geq t$$
$$BA\nabla J(a,b), < t \quad (2)$$

In which, ∇ is a representation of the gradient operator, N refers to the commotion and BA symbolises the surrounding region. The images that have been categorised as noise are subjected to the filtering modification once each pixel has been classified. After updating, the pre-processed images are delivered to the feature extraction stage.

Feature Extraction Using the Center and Intrinsic Moment Methods

Gaining critical picture features is made possible by applying centroid and invariant moment extraction methods. Employing these methods for image analysis frequently results in resilient and immensely valuable features. Centroid features help us find the exact middle of an object, while invariant moments give us details about the object's shape and how it's spread out, without changing even if the object is moved or turned. By combining these two methods, we can get both the position and shape details of objects, which makes it much easier to recognize and classify them correctly. This technique is really useful in a lot of different areas, like looking closely at medical images, keeping track of objects in videos, or identifying faces. It's all about pulling out the right details from images to make sure we get accurate results.

Technique for Obtaining Centroid Characteristics

In deep learning and working with images, finding the "centroid" is a really useful trick. It's like figuring out the main gathering point or the average spot in a bunch of data. When it comes to images, this method helps us find where exactly an object is sitting. The centroid is basically the spot that sits right in the middle of all the parts of an object, found by averaging out the places of all its pieces. This is super handy for understanding more about where things are in pictures. If a document contains N discrete points $Y_1 \ldots . Y_N$, next, the centroid (C_E) is decided by Equation (3):

$$C_e = \frac{1}{N}\sum_{j=1}^{N} Y_j \quad (3)$$

Shapes in image processing and computer vision are essentially composed of pixels, and the centroid is the weighted average of all the pixels that make up the shape.

Open CV's moment feature aids in locating the center of a blob. A clear comprehension of the Image Moment holds significant importance in this context. Image Moment, functioning as a weighted average of pixel intensities aids in determining specific image attributes like radius, area, and centroid. Typically, developers convert the image into a binary format before commencing centroid identification. Below, equations (4) and (5) show how to calculate the centroid:

$$C_{\mathrm{a}} = \frac{m_{10}}{m_{00}} \quad (4)$$

$$C_{\mathrm{b}} = \frac{m_{01}}{m_{00}} \quad (5)$$

In which, C_{a} symbolizes the a centroid's coordinate, C_b symbolizes the b centroid's coordinate, and m symbolizes the current. How to get the centroid of a blob using Open CV will also pinpoint the center of the blob by performing the following activities:

- Make the picture grayscale.
- A reduction in the image size.
- Once the seconds have been calculated, locate the image's middle.

Techniques for Extracting Invariant Moment Features

In computer vision and image processing, invariant moment feature extraction is frequently used to obtain image attributes. Moments are mathematical functions that describe how items are arranged and shaped in images. Computing a sequence of moments that are resistant to specific transformations, including scaling, translation, and rotation, is necessary to extract invariant moment properties. For example, moment invariants

are used to extract detailed characteristics for shape recognition and identification analysis. This method is widely used in many disciplines, especially in tasks related to recognition because of its efficacy in producing feature vectors that accurately represent a picture.

"Invariant" is the word (I_v) is used to describe an image's (or function's) property that remains constant or only slightly varies as the image is modified.

When $f(a,b)$ is the initial picture and D_0 symbolizes the transformer function of the picture, or deterioration operator, wherein a then b are the coordinates of the image's pixels and f result is the intensity of each pixel, and

$$J(f) \approx J(D_0(f)) \tag{6}$$

Equation (7) is commonly used to express an invariant as a vector,

$$J = (J_1, J_2, ..., J_n) \tag{7}$$

Here, 'n' represents the number of edits made to the picture. Using equation (6) across voids, mapping each invariant vector from the image dataset should ideally result in the identification of discrete clusters representing diverse objects. This occurs due to the significant divergence in values for I among the various objects.

The Moment Invariant technique primarily captures the structural attributes of binary images. This technique is closely related to moments that are calculated from binary images, which are also known as silhouette moments. The universal calculation of any instant type of order is described by Equation (8), γ besides δ a function measuring picture intensity $f(a,b)$ of $n \times m$ size of bits.

$$M_{\gamma\delta} = N_f \sum_{a=0}^{a-1} \sum_{b=0}^{b-1} \prod_{\gamma\delta}(a,b) f(a,b) \tag{8}$$

In which, N_f stands for the factor of normalization, $\prod_{\gamma\delta}$ symbolizes the times Core, γ and also δ consists of the specific polynomial of the orthogonal basis and its product. By classifying the most recent Moment family based on the characteristics of Kernel's polynomials, many kinds of moments are produced. As a part of what's known as "Moment Invariants," 2-D geometric moments characterize the distribution function within an image. These moments assist in establishing orthogonal invariants concerning linear transformations, including factors such as translation, scaling, and rotation, through a collection of moment invariants rooted in algebraic function theory. This function is also referred to as the Geometric Moment Invariant (GMI). The GMI function's general definition $M_{\gamma\delta}$ of the directive $r+s$ regarding the continuous function in two dimensions $f(a,b)$ intended for $r, s = 0,1,2,3....$ is stated in Equation (9).

$$M_{\gamma\delta} = \int_{a=1}^{n} \int_{b=0}^{m} a^{\gamma} b^{\delta}(a,b) \tag{9}$$

When it comes to the study of digital images, Equation (9) can be expressed as (10);

$$M_{\gamma\delta} = \sum_{a=1}^{n} \sum_{b=0}^{m} a^{\gamma} b^{\delta}(a,b) \tag{10}$$

In which $f(a,b)$ indicates the value of a pixel in a size picture nxm . The central moments of equation ten can be determined using Equation (11);

$$\mu_{\gamma\delta} = \sum_{a=1}^{n} \sum_{b=0}^{m} \left(a - \bar{a}\right)^{\gamma} \left(b - \bar{b}\right)^{\delta} f(a,b) \tag{11}$$

Anywhere, $\bar{a} = \frac{M_{10}}{M_{00}}$ and $\bar{b} = \frac{M_{01}}{M_{00}}$. The scaling factor in Equation (12) can then be used to derive the invariant qualities for normalising the central moment.

$$\eta_{\gamma\delta} = \frac{\mu_{\gamma\delta}}{\mu_{00}^{\frac{r+s+2}{2}}} \tag{12}$$

The six invariants to translation, rotation, and scaling functions are defined by equation (13).

$$\begin{aligned}
\varphi_1^{'} &= \eta_{20} + \eta_{02} \\
\varphi_2^{'} &= (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \\
\varphi_3^{'} &= (\eta_{30} + 3\eta_{12})^2 + (3\eta_{21} + \eta_{03})^2 \\
\varphi_4^{'} &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
\varphi_5^{'} &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left((\eta_{30} + 3\eta_{12})^2 - (3\eta_{21} + \eta_{03})^2\right) + (3\eta_{21} + \eta_{03})(\eta_{21} + \eta_{03})\left(3(\eta_{30} + \eta_{12})^2 - (\eta_{12} + \eta_{03})^2\right) \\
\varphi_6^{'} &= (\eta_{20} - \eta_{02})\left((\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right) = 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})
\end{aligned} \tag{13}$$

These seven moments (13) could be regarded as the seven characteristics of the invariant moment. The gathered attributes are then updated to the classification layer utilising EDBN in order to classify scalp detection.

Categorization of Enhanced Belief Neural (EDBN) Systems

Using enhanced deep belief neural networks, or EDBNs, is how the scalp is identified in the scalp detection method. This section provides an introduction to EDBNs, detailing the construction of an EDBN-based classifier. Researchers utilize EDBNs to generate deep features from file types, serving as effective feature extractors. EDBNs facilitate the representation of underlying data through a systematic layer-by-layer supervised learning approach. These networks are rooted in a base structure known as the Restricted Boltzmann Machine (RBM), functioning as a neural network comprised of two layers—one exposed and one hidden. Figure 2 illustrates the structural arrangement of an EDBN.

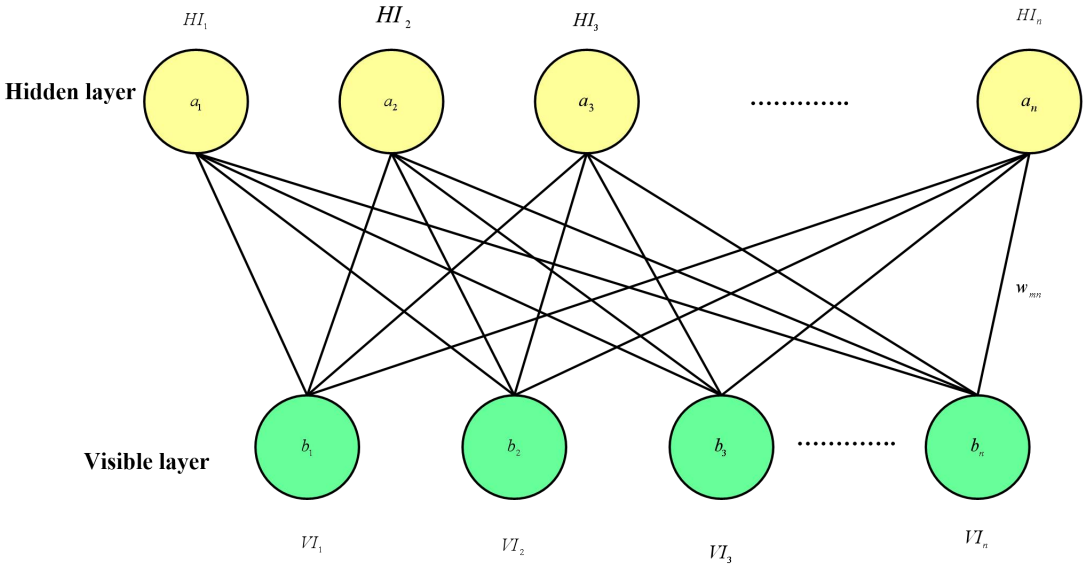


Figure 2. Composition of EDBN

Within a layer of an RBM (Restricted Boltzmann Machine), units are not directly connected to one another. Instead, connections exist between the visible and hidden layers through symmetric weights. The hidden units are pivotal in establishing meaningful correlations observed in the visible units. The weights of an RBM are found using a greedy layer-wise pretraining technique, which makes weight initialization easier. Using unlabeled data, this unsupervised approach of data preparation defines the generative weights of an RBM. Equation 14 uses the weights of the RBM for the visible and hidden layers to represent the energy of a joint configuration.

$$\text{Envi,hi;}\theta\text{=-n=1VIm=1HIvishitWst-n=1VIbnvis-m=1HIamhis} \tag{14}$$

Somewhere, θ = {W, b, a}, vis, the binary state of the hidden t and visible s, vis, hit ϵ 0,1, W stands for the symmetric weight parameters with VI×HI dimensions, The visible number units are denoted by VI, The bias parameters of visible units are represented by b, the bias parameters of hidden units are represented by a, and the number of hidden units is represented by HI.

The equation shows how each possible visible-hidden vector combination in an RBM is given a probability by the energy function (15).

$$pr(vi,hi) \propto exp(-Evi,hi) \quad (15)$$

Equation 16 gives the derivative of a visible vector's log-likelihood with respect to the weights.

$$\partial \log P(v) \partial wji = via,hibdata - via,hibmodel \quad (16)$$

The expectation with respect to the given distribution indicated in the subscript is represented by the angle brackets. 'via, hibdata' signifies the expectation pertaining to the training data, while 'via, hibmodel' denotes the expectation linked to the model. The gradient of Equation (17)'s log, with 'ε' representing the learning rate, forms the foundation for the weight update rule.

$$\Delta wmn = \varepsilon via,hibdata - via,hibmodel \quad (17)$$

Contrastive divergence learning approximates the gradient Δwmn. The binary states of the visible units are calculated by equation (19), and the binary states of the hidden units are computed by equation (18). The original observable vector is recreated by the computed visible states. This learning rule offers a practical means of estimating the gradient from the training data, proving valuable and adequate in identifying optimal features.

$$phit = 1 = 11 + exp(-ut - sviswmn) \quad (18)$$

$$pvis = 1 = 11 + exp(-xs - thitwnm) \quad (19)$$

It's evident that reaching optimal acquisition requires only a single step, significantly diminishing training duration. Ultimately, the developed model successfully classifies conditions like oily hair, folliculitis, dandruff, and hair loss in scalp detection.

Grey Wolf Optimisation (GWO) Method of Optimisation

Grey Wolf Optimisation (GWO) is a technique that addresses optimisation problems across multiple domains, such optimising parameters for Enhanced Deep Belief Networks (EDBNs). It is inspired by the social structure and hunting tactics of grey wolves. EDBNs encompass multiple parameters crucial for achieving peak performance, and employing GWO proves efficient in optimizing these parameters, thereby enhancing the network's effectiveness. This algorithm works like grey wolves hunting together. In a wolf pack, the alpha, beta, and delta wolves lead the group. They're the top wolves that guide everyone else during the hunt by changing their locations and actions, following some rules that sort of mimic how real wolves behave. Even the omega wolf, which might not be as strong or influential, plays a role by helping find food. There's a diagram (Figure 3) that shows how this wolf-inspired strategy is used to tweak the settings of an Enhanced Deep Belief Neural (EDBN) network to make it work better. Then, there's a step-by-step guide that explains exactly how to use this Grey Wolf Optimization (GWO) method to adjust the network for better performance.

Convergence of the Optimization Algorithm

In this research, making sure the algorithm zeroes in on the right answers efficiently is key to correctly spotting and diagnosing hair and scalp issues. We use a cool technique inspired by how grey wolves hunt, called Grey Wolf Optimization (GWO), to fine-tune the settings of our Enhanced Deep Belief Neural (EDBN) network. The idea of convergence here means getting to a point where our algorithm has adjusted the EDBN settings just right, so it can accurately identify problems by looking at specific features. This fine-tuning process keeps on going, making little adjustments until it's clear that the settings are as good as they're going to get, based on some specific stopping points we've set. These could be hitting a certain goal, not seeing much change from one step to the next, or simply running out of time to make more changes.These are discussed below. Also, the experimental validation plays a crucial role in evaluating the convergence behaviour of the GW-EDBN algorithm. Monitoring performance metrics such as classification Precision, Recall, F-measure and time spent on execution provides insights into the algorithm's convergence speed and the quality of the solution obtained. All these points are discussed below.

Stage 1: Initialization

Commence by establishing the initial population of wolves to optimize the parameters pr (vi, hi) and Evi, hi within the EDBN classifier method, aimed at improving classification efficiency. Subsequently, initialize the position of each wolf according to Equations (20) and (21).

$$L^{'} = \left| F \times P_{y}(ct) - P(t+1) \right| \quad (20)$$

$$P(t+1) = P_{y}(\mathrm{ct}) - H \times L^{'} \quad (21)$$

Wherever, ct symbolizes the iteration that is being used currently, F then H depicts the vectors of coefficients, P_{y} is a vector representing the prey's position, also $L^{'}$ depicts the position vector of the grey wolf.

Stage 2: Haphazard creation

The starting places of the wolves are chosen at random within the allocated search area. By standardising the cumulative total of all iterations, equation (22) controls the wolf's random walk.

$$H = 2\mathrm{h} * R - h * H * \quad (22)$$

Thus, R provides the random vectors with elements of in the interval [0,1] h decreasing straight over the span from 2 to 0.

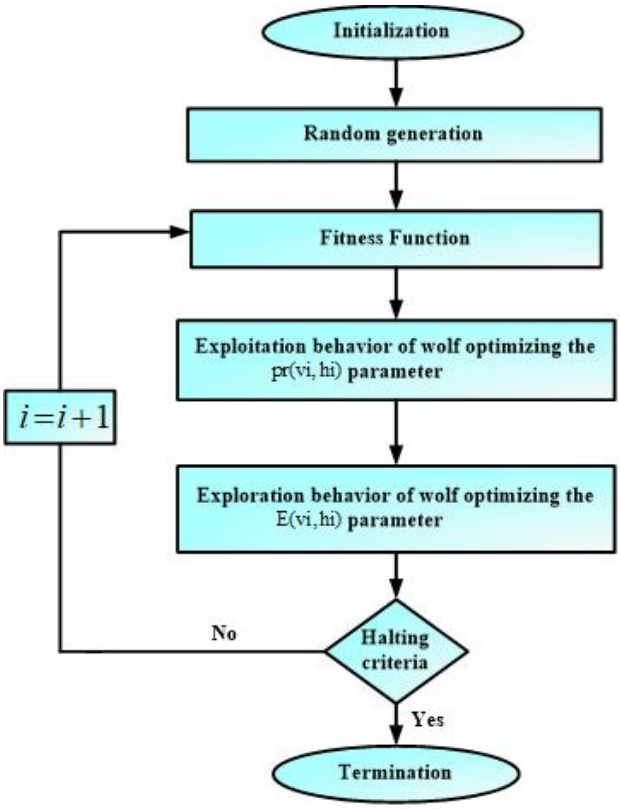


Figure 3. GW Flow Schematic for EDBN Classifier Parameter Optimization

Step 3: Function of fitness for optimizing EDBN parameters

The optimal wolf location is chosen to generate the function in this fitness $H = 2\mathrm{h} * R - h * H *$ adjusting the EDBN classifier's pr (vi, hi) and Evi, hi parameters; equation (23), which represents the fitness function of the classifier,

Fitness function= optpr (vi, hi) and Evi, hi (23)

Step 4: Wolf behavior is exploited, and the pr (vi, hi) parameter is optimized

Grey wolves conclude their hunt by incapacitating the prey, compelling it to halt. In GWO computations, the objective is to reposition themselves akin to the alpha, beta, & delta zones to approach the prey. Nevertheless, the GWO analysis tends to favour local solutions, leading to stagnation for its leaders. While acknowledging the value of the entire spectrum, GWO necessitates leaders to guide the exploration. Optimizing the prvi,hi parameter is achieved through this exploitative behaviour.

$$L_{\alpha}^{'} = \left|F_1 \times F_{\alpha} - P\right| + pr(vi, hi)$$
$$L_{\beta}^{'} = \left|F_2 \times P_{\beta} - p\right| + pr(vi, hi) \quad (24)$$
$$L_{\delta}^{'} = \left|F_3 \times P_{\delta} - P\right| + pr(vi, hi)$$

Step 5: Wolf exploration behavior and Evi, hi parameter optimization

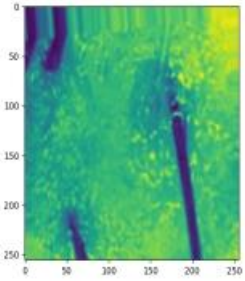
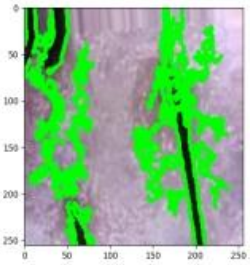
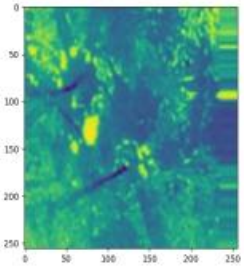
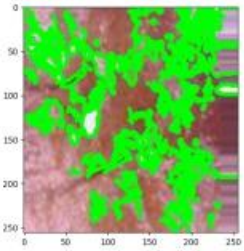
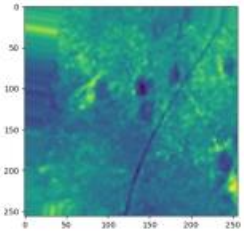
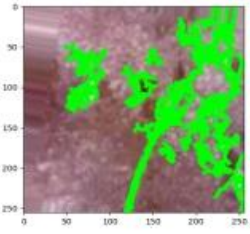
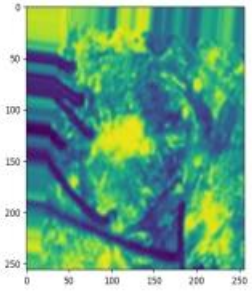
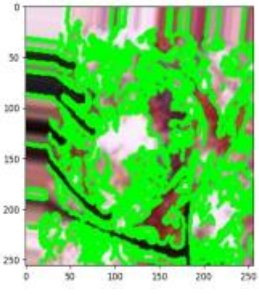
Different alpha, beta, and delta tiers are traversed by grey wolves according to their social hierarchy within the pack. They disperse to hunt for prey and then regroup to pursue the objective. Similar to this, the GWO method exhibits evolutionary characteristics when optimising, giving exploration of the search space priority over local optimum solutions.

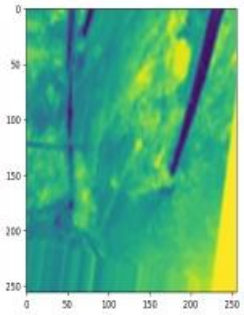
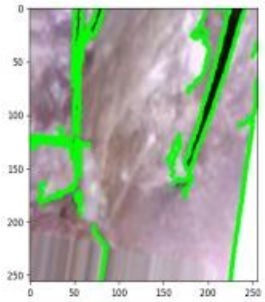
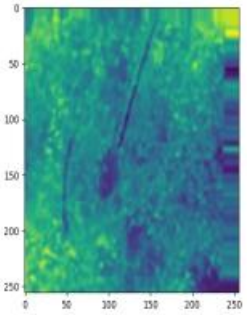
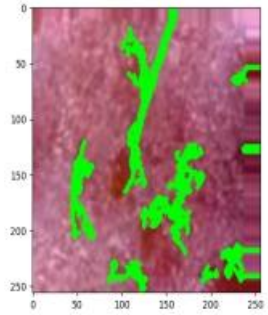
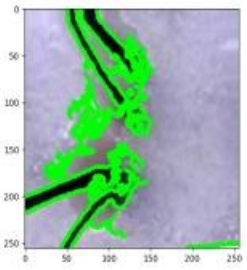
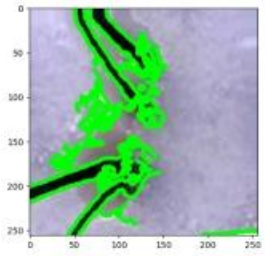
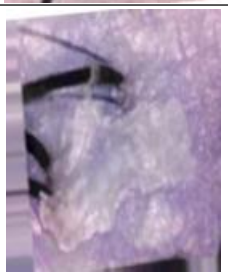
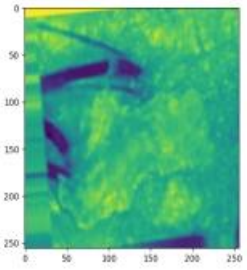
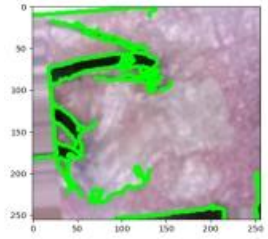
$$P = P_{\alpha\beta\delta} - H \times \left(L_{\alpha\beta\delta}^{'}\right) + E(vi, hi) \quad (25)$$

Step 6: Removal

The EDBN factors benefit from GWO's combination of exploration and exploitation behaviours. This enhancement reduces computation time, accommodating errors, and improving accuracy in the resulting objective function. The algorithm's third phase iterates until the termination criteria are met. The initial input, processed in advance and feature-extracted images of the created model are shown in Table 2.

Table 2. Pictures of the Created Model that have been input, preprocessed, and feature extracted

Image Input	Image Preprocessed	Image with Features Extracted	Arrangement
			Oily hair
			Folliculitis
			Hair loss
			Folliculitis

			Dandruff
			Hair loss
			Oily hair
			Dandruff

RESULTS AND DISCUSSION

Various performance metrics are utilized during experimental evaluations to gauge the proposed approach against established models. Python-based tools are used to simulate these experimental results. The initially gathered dataset undergoes pre-processing to remove any distortions and extract essential features using centroid and invariant moment techniques. Following that, the EBDN conducts scalp detection while optimizing its parameters using GWO, leading to a significant enhancement of the scalp detection system. Figure 4 portrays the confusion matrix and ROC characteristics of the developed model.

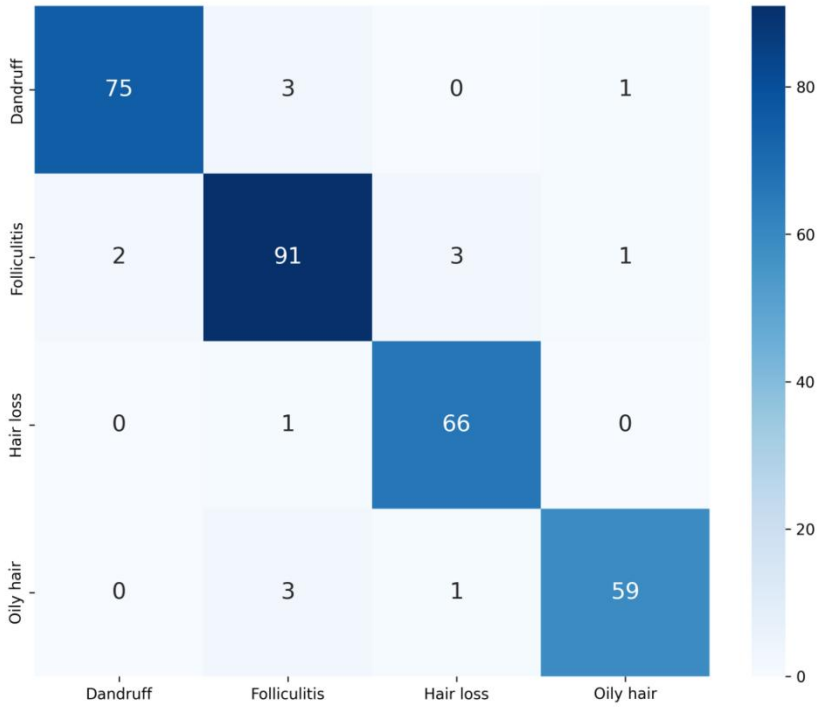


Figure 4. (a) Metrics of Performance

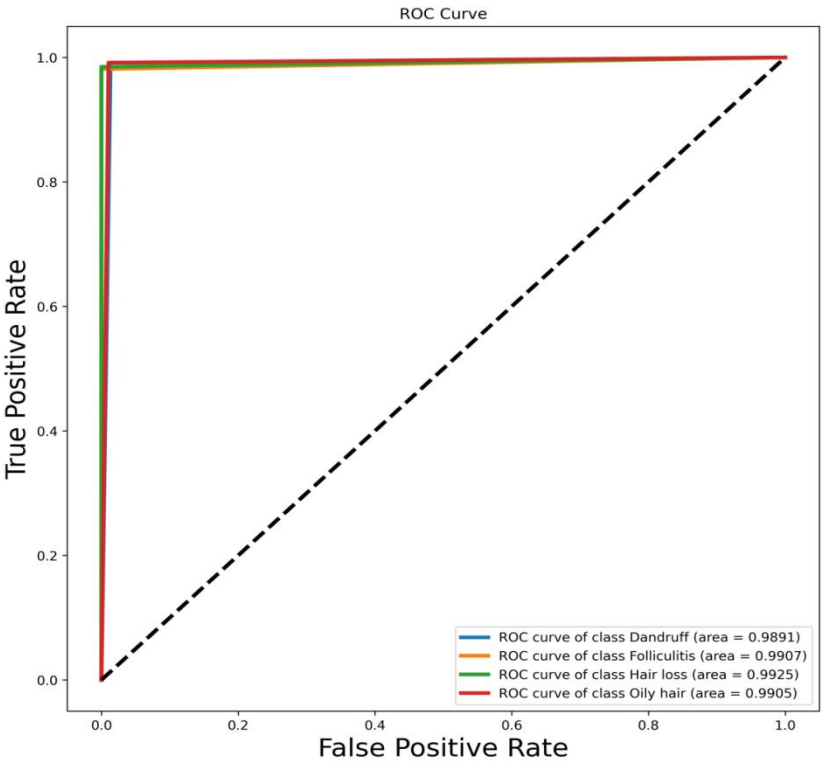


Figure 4. (b) Features of ROC

Figure 4. ROC Features and the Confusion Matrix

Description of the Dataset

I collected and trained the system using scalp hair images and SEM images. These images exhibit symptoms like oily hair, folliculitis, dandruff, and hair loss, totalling 1820 in number. The training phase involved 1456 images, while 364 images were used for testing purposes.

Analysis of Performance

The developed model's effectiveness was validated against established methods, evaluating accuracy, precision, recall, F-measure, and execution time. Furthermore, the performance of the hair scalp diagnostic system was compared to conventional models such as DL-CNN [21], Scalp Eye [22], and Norwood-Hamilton Technique (NHT) [24].

Performance parameters:

Precision: Precision measures the proportion of correctly predicted positive cases out of all cases predicted as positive.

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{26}$$

Recall: Recall measures the proportion of correctly predicted positive cases out of all actual positive cases.

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{27}$$

Accuracy: Accuracy measures the proportion of correctly predicted cases out of all cases.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{28}$$

Specificity: Specificity measures the proportion of correctly predicted negative cases out of all actual negative cases.

$$\text{Specificity} = \frac{\text{TN}}{(\text{TN} + \text{FP})} \tag{29}$$

F1-Score (F-Measure): The F1-score is the harmonic mean of precision and recall, providing a balance between them.

$$\text{F1-Score} = 2 * \frac{\text{Precision} * \text{Recall}}{(\text{Precision} + \text{Recall})} \tag{30}$$

Error Rate: Error rate calculates the proportion of incorrect predictions out of all predictions.

$$\text{Error Rate} = \frac{\text{FP} + \text{FN}}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{31}$$

Where TP (True Positives) represents the number of correctly predicted positive instances while TN refer to True Negatives represents the number of correctly predicted negative instances. Furthermore, FP and FN referred to False Positives and False Negatives representing the number of incorrectly predicted positive and negative instances respectively.

Execution Time: Execution time refers to the time taken by a system or algorithm to complete a task or process.

Time complexity is typically denoted using Big O notation, such as O(n), O(n^2), etc., where O(1) is the Constant time complexity, indicating that the algorithm's execution time remains constant regardless of input size. While O(n) is linear time complexity, indicating that the algorithm's execution time increases linearly with the size of the input.

Using Grey Wolf Optimization (GWO) to fine-tune the settings of Enhanced Deep Belief Networks (EDBN) helps the system get better at spotting scalp issues. GWO is like a smart assistant that adjusts the network's settings—things like weights, biases, and how fast it learns—to make sure it's working as well as it can. This not only makes the system more accurate at detecting problems but also helps avoid getting stuck on solutions that aren't the best (overfitting). By searching globally for the best settings, GWO makes sure that the network is really getting a thorough adjustment, which means it's better at adapting to different kinds of scalp conditions. Plus, with GWO's help, the EDBN becomes quicker and more reliable at learning from scalp images, leading to faster and more trustworthy results in identifying scalp issues.

While the primary focus of this contribution lies within the context of hair and scalp damage detection, the techniques and methodologies employed, particularly the application of Grey Wolf Optimization (GWO) to enhance the effectiveness of Deep Belief Neural Networks (DBNN) for scalp detection, hold significant potential for broader applications beyond hair care and scalp assessment. The optimization framework

demonstrated in this study can be adapted and extended to various domains requiring image analysis, pattern recognition, and anomaly detection. Industries such as medical imaging, surveillance, agriculture, and manufacturing can benefit from the robustness and efficiency of the proposed methodology for detecting anomalies, defects, or abnormalities in images and visual data, thus demonstrating the generalizability and versatility of the contribution beyond the specific domain of hair care and scalp assessment.

For detailed information about the complexity or architecture of the compared conventional models like DL-CNN, Scalp Eye, and Norwood-Hamilton Technique (NHT)), it is much more difficult and challenging to clarify the differences in performance accurately. To address this challenge, we adopt an approach centred on empirical performance metrics. Through the evaluation of key metrics such as accuracy, precision, recall, and F1-score, along with additional indicators like specificity, error rate, and execution time, we gain a holistic understanding of the relative efficacy of the proposed method compared to existing models. These metrics furnish quantitative measures across various facets of model performance, facilitating an objective appraisal and comparison. By leveraging these performance indicators for established models like DL-CNN, Scalp Eye, and NHT, we distinguish the restraints of model performance. Thus it highlights the superiority of our proposed approach despite the absence of detailed architectural insights. This methodology ensures a robust evaluation framework, effectively tackling the challenge presented by the limited architectural information available for conventional models.

Accuracy

The accuracy of the designed model underwent comparison with DL-CNN, Scalp Eye, and NHT models. Over different epochs, DL-CNN, NHT, and Scalp Eye achieved varying accuracies: 88%, 98.25%, and 92.5% at 20 epochs; 86.7%, 96%, and 91% at 40 epochs; 84%, 94.2%, and 89.4% at 60 epochs; 82.12%, 92%, and 87.34% at 80 epochs; and 80%, 91.3%, and 85% at 100 epochs, respectively.

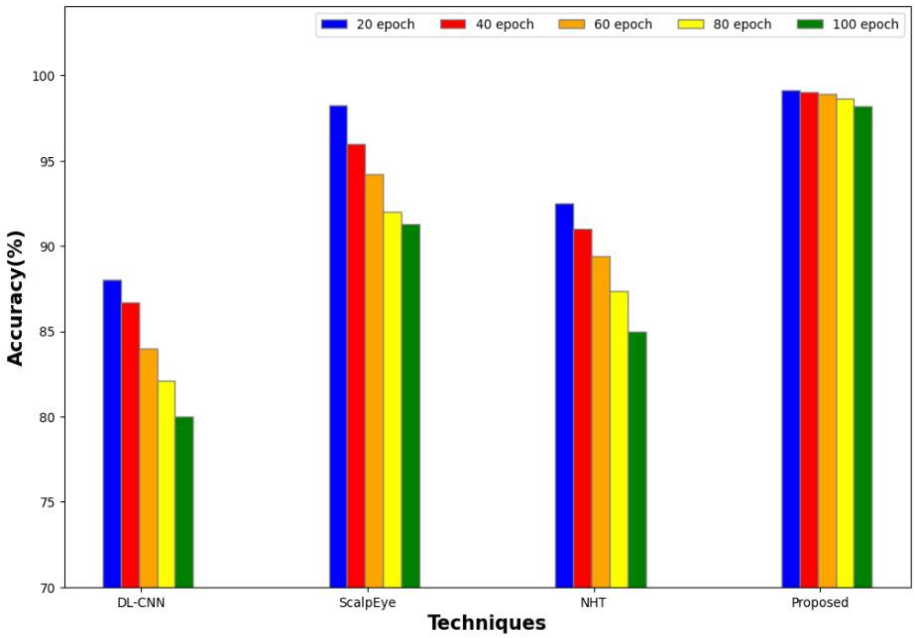


Figure 5. Comparison of Accuracy

Figure 5 displays the accuracy comparison between the designed model and conventional models. In contrast to other models, the designed one showcases exceptional abilities in identifying and detecting hair and scalp issues. Across different epochs, the developed model achieves accuracy scores of 99.12%, 99.02%, 98.91%, 98.66%, and 98.22% at 20, 40, 60, 80, and 100 epochs, respectively.

Precision

The precision of the designed model was validated against DL-CNN, Scalp Eye, and NHT models. Across different epochs, DL-CNN, Scalp Eye, and NHT achieved precision scores of 87.5%, 98%, and 93% at 20 epochs; 86%, 95.7%, and 90.12% at 40 epochs; 83.6%, 93.5%, and 88% at 60 epochs; 82%, 92%, and 86.6% at 80 epochs; and 79.5%, 90.6%, and 84% at 100 epochs, respectively.

Figure 6 exhibits the precision comparison between the designed model and traditional models. Unlike other models, the designed model showcases superior accuracy in making precise predictions or retrieving pertinent data. Over various epochs, the developed model reaches precision scores of 99%, 98.92%, 98.78%, 98.56%, and 98.12% at 20, 40, 60, 80, and 100 epochs, respectively.

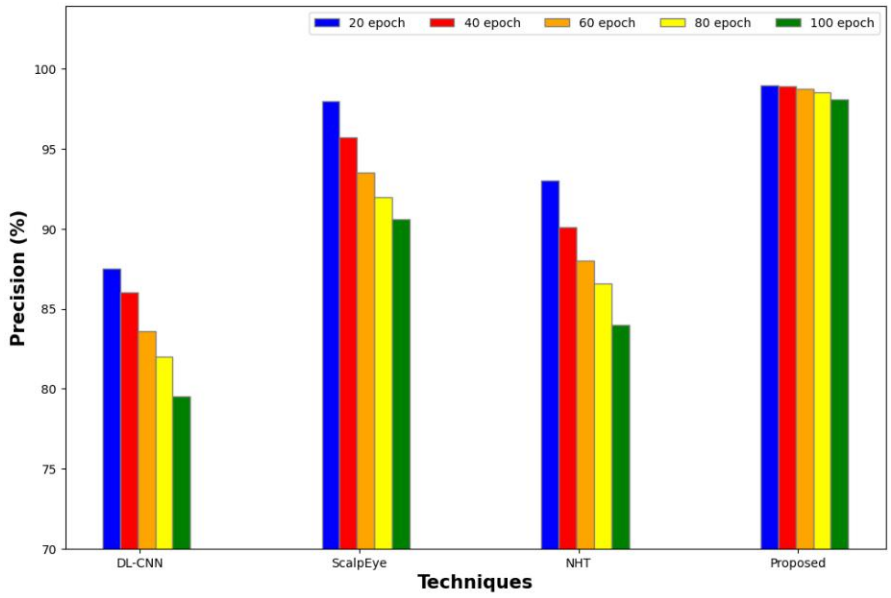


Figure 6. Comparison of Precision

Recall

The designed model's recall was assessed in comparison to DL-CNN, Scalp Eye, and NHT models. Over various epochs, DL-CNN, NHT, and Scalp Eye achieved recall scores of 87.3%, 96%, and 91.6% at 20 epochs; 86%, 94.6%, and 90% at 40 epochs; 83.3%, 92%, and 88% at 60 epochs; 81%, 90.3%, and 86.9% at 80 epochs; and 79.5%, 88.5%, and 84.3% at 100 epochs, respectively.

Figure 7 displays the recall comparison between the designed model and traditional models. Unlike other models, the designed one stands out in precise detection of hair and scalp issues. Over various epochs, the developed model attains recall scores of 99.09%, 99.02%, 98.87%, 98.54%, and 98.18% at 20, 40, 60, 80, and 100 epochs, respectively.

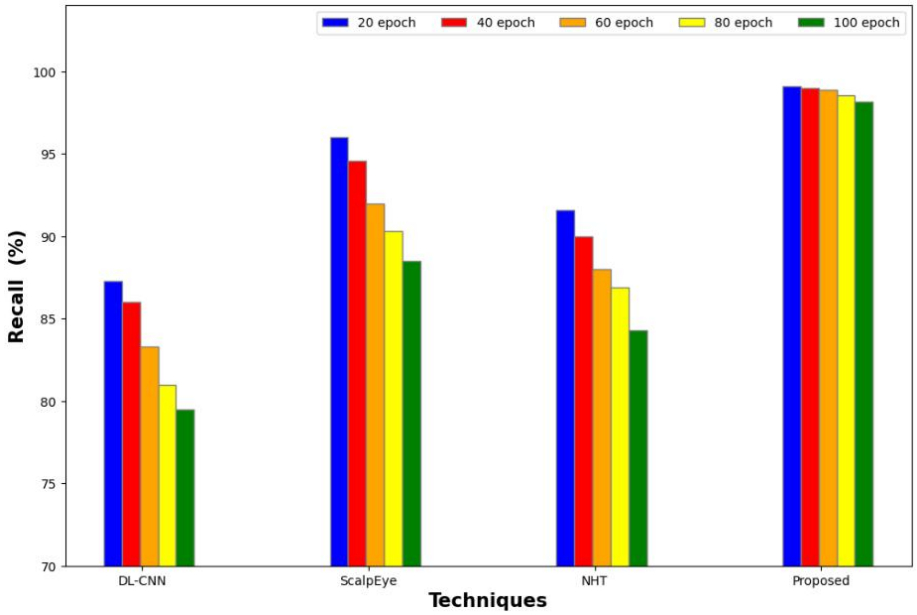


Figure 7. Comparison of Recall

The F-Measure

The F-measure of the designed model was validated against DL-CNN, Scalp Eye, and NHT models. Across various epochs, the F-measure scores for DL-CNN, NHT, and Scalp Eye were as follows: 85.5%, 97.4%, and 91% at 20 epochs; 84%, 95.4%, and 90.3% at 40 epochs; 82.5%, 93%, and 88% at 60 epochs; 81%, 91.8%, and 86.53% at 80 epochs; and 79.21%, 90.04%, and 84.4% at 100 epochs, respectively.

Figure 8 presents the F-measure comparison between the designed model and conventional models. Compared to other models, the developed model excels, achieving F-measure scores of 99%, 98.83%, 98.55%, 98.24%, and 98% at 20, 40, 60, 80, and 100 epochs, respectively.

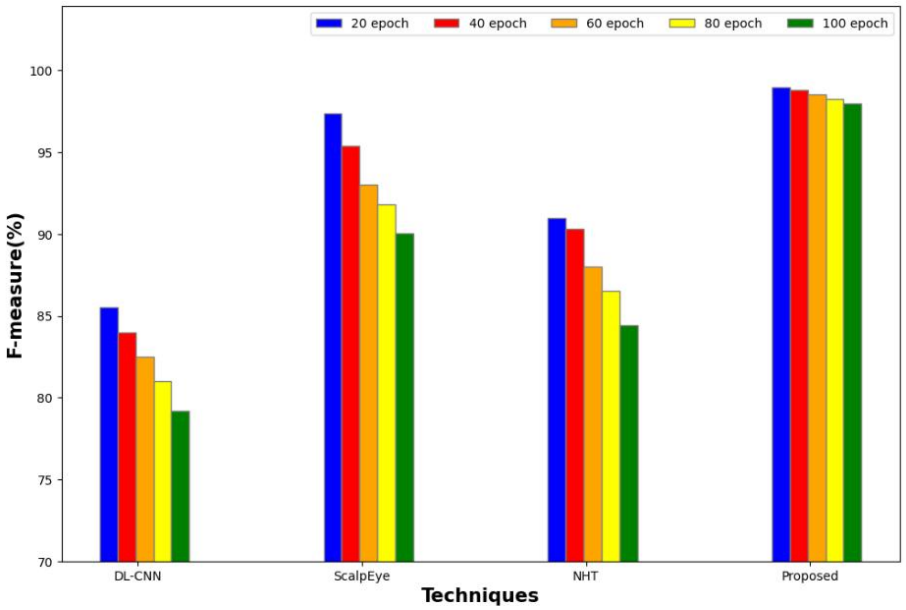


Figure 8. Comparison of F-Measures

Time Spent on Execution

The comparison of execution times between the designed model and DL-CNN, Scalp Eye, and NHT models was conducted. At various epochs, DL-CNN, NHT, and Scalp Eye took the following durations: 18s, 25s, and 12s at 20 epochs; 20s, 27s, and 15s at 40 epochs; 23s, 30s, and 18s at 60 epochs; 28s, 32s, and 21s at 80 epochs; and 33s, 38s, and 25s at 100 epochs, respectively.

In Figure 9, the comparison of execution times between the designed model and traditional models is presented. The designed model outperforms others by detecting hair and scalp issues more rapidly. Specifically, across epochs, the developed model takes 4s, 6s, 9s, and 11s for 20, 40, 60, 80, and 100 epochs, respectively, illustrating its efficiency.

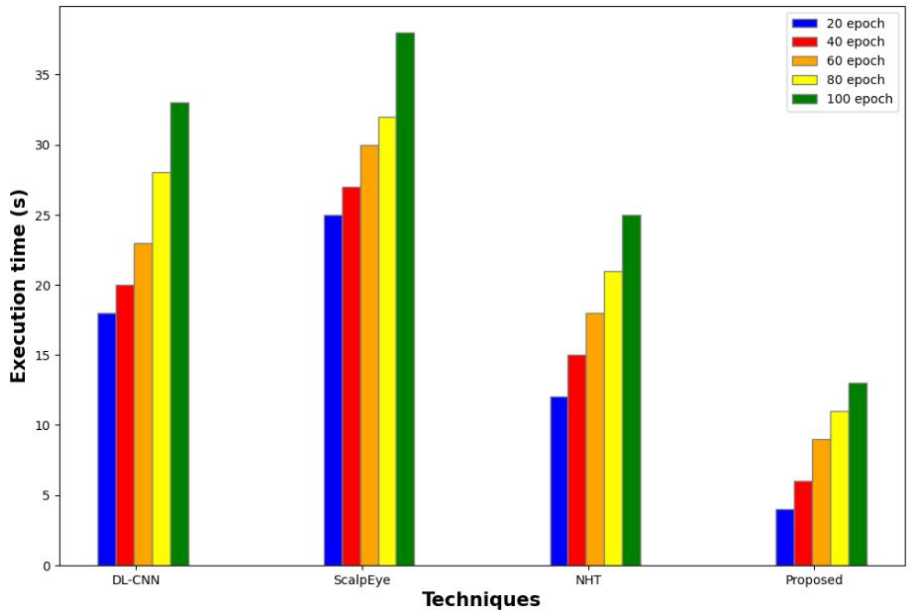


Figure 9. Comparison of Execution Times

Time Complexity Analysis

Execution time showcases the real-world performance of different methods in processing data. Lower values of execution time indicate quicker execution and more efficient algorithms. The GW-EDBN method we came up with is really quick, taking only 3 time units to do its job, which shows it's a lot faster than the others. For example, another method called NHT takes the longest, about 9 time units, which means it's not as quick. These times help us understand how fast or slow each method is, basically telling us how efficiently they work. So, when we look at how good these methods are, it's important to think about how fast they can do their work, along with other things that show us how well they perform.

Our findings showed that using more filters (32 instead of 16) made the training take about 14% longer per round. It also was a bit slower than another model we checked out, Model E, by about 6.1%. However, this extra time wasn't for nothing—the setup with 32 filters ended up being more accurate, hitting an accuracy rate of 0.87 after going through the eight data sets. When we charted the accuracy against the time taken for each training session, we noticed that models with more filters not only got better over time but also showed a clear jump in accuracy after the third batch of data.

So, this dive into the numbers helps us understand the balance between how long you're willing to wait for your model to train and how accurate you want it to be. It highlights the importance of picking the right setup depending on what's more critical for your needs—speed or precision.

Discussion

The performance of the GW-EDBN model was commendable, showcasing exceptional results in Recall, f1-Score, precision, accuracy, and runtime. Consequently, this scheme successfully eradicated initial training deficiencies. Thus, the advanced GW-EDBN technique significantly enhances scalp detection performance.

Table 3. Overall Performance Metrics

Methods	Performance Assessment				
	Precision	F1-Score	Accuracy	Recall	Execution Time
DL-CNN	99	99	99.12	99.09	4
Scalp Eye	98.92	98.83	99.02	99.02	6
NHT	98.78	98.55	98.91	98.87	9
GW-EDBN (Proposed)	99.09	99.08	99.68	99.89	3

The comprehensive performance metrics are tabulated in Table 3, revealing that the proposed GW-EDBN model excels in all parameter validations. Remarkably, it achieves outstanding scores of 99.89% in recall, 99.09% in precision, 99.08% in F1-Score, and 99.68% in accuracy. These results strongly support the reliability of the GW-EDBN method, confirming its proficiency in detecting scalp issues. Furthermore, Figure 10 demonstrates the accuracy and loss trends across epochs for both the training and testing datasets.

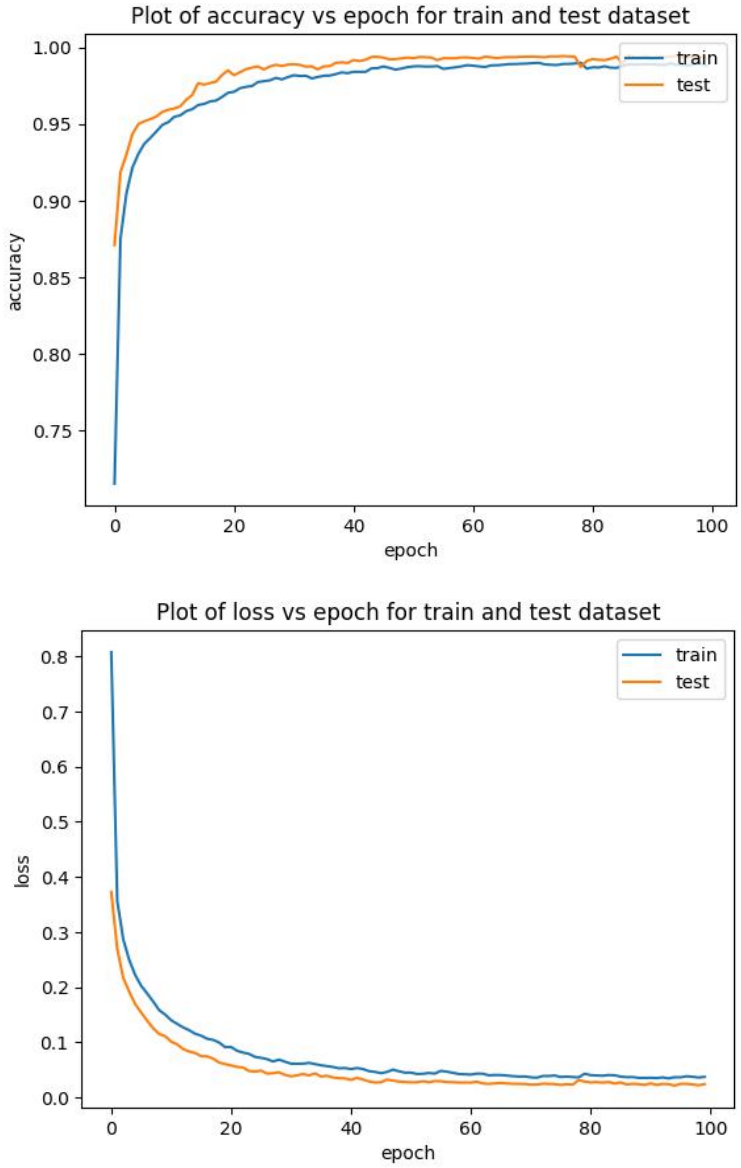


Figure 10. Training and Testing Datasets, Accuracy and Loss v/s Epoch

The performance comparison of various optimization techniques, including Grey Wolf Optimization (GWO), Genetic Algorithms (GA), Improved PSO (IPSO), Particle Swarm Optimization (PSO), and Differential Evolution (DE), reveals insights into their effectiveness based on mean values, standard deviations, and ranks. GWO achieves the highest performance with the lowest mean value of 88.58 and a very low standard deviation of 0.025, indicating high precision and consistency in its results. It holds the top rank among the listed techniques, demonstrating its efficiency in optimization tasks (Table 4).

Table 4. Comparison of Optimization Techniques

Parameter	Mean	Std	Rank
GWO	88.58	0.025	1
GA	89.48	2.402	2
IPSO	89.86	3.914	3
PSO	90.79	3.424	4
DE	93.62	6.735	5

However, GA follows closely behind GWO with a mean value of 89.48 and a standard deviation of 2.402. While GA performs slightly better in terms of mean value compared to GWO, its higher standard deviation suggests slightly more variability in its results, leading to a lower rank compared to GWO. An Improved PSO exhibits a mean value of 89.86 and a standard deviation of 3.914. Despite having a higher mean value than GA, its larger standard deviation indicates more variability in its performance, resulting in a lower rank compared to GA. Furthermore, PSO achieves a mean value of 90.79 and a standard deviation of 3.424. While PSO has a higher mean value than IPSO, its slightly higher standard deviation leads to a lower rank compared to IPSO. DE shows the highest mean value of 93.62 among the listed techniques but has a significantly larger standard deviation of 6.735. This indicates a wider spread of results and higher variability compared to other techniques, resulting in the lowest rank among the listed methods. While DE has the highest mean value, its larger standard deviation indicates less consistency in performance compared to GWO, which demonstrates the most stable and efficient optimization performance among the listed techniques.

The study achieves high recall and precision rates. Table 3 shows overall performance metrics, other methods like DL-CNN, Scalp Eye, and NHT have also demonstrated high precision and recall rates in the range of 98 to 99 per cent. The proposed GW-EDBN method has achieved results in a similar range, indicating its competitiveness with existing methods. However, concerns about potential over fitting due to the limited size and diversity of the dataset are valid. To tackle the problem of the model being too specific and not performing well on new, unseen data (a common issue known as overfitting), this study uses special techniques to adjust the model's settings. By tweaking these settings, we aim to strike a good balance: we want the model to be accurate on the data it's trained on but also flexible enough to handle new data it hasn't seen before. According to the results shown in Table 3, our approach not only holds its ground when compared to other methods but also shows some improvements. So, even though our data isn't perfect and there are concerns about overfitting because of how the data might not be varied enough, these adjustments to the model's settings help reduce that risk.

CONCLUSION

In this study, we introduce a smart system called GW-EDBN that's great at checking for scalp problems using pictures from the internet. It focuses on spotting oily hair, infections, dandruff, and hair loss. To make sure the pictures are clear, we use a cleaning-up process called AGF. Then, we use special techniques to pick out important details from these pictures. These details help the system figure out what's wrong with the scalp. We also use a method inspired by how grey wolves work together to make our system even smarter at finding these problems. Our tests, done on a computer, show that GW-EDBN is really good at this, even better than some older methods, with nearly perfect scores in being right and precise.

REFERENCES

[1] A. Shariat, A. Zarei, S. A. Karvigh, and B. M. Asl, "Automatic detection of epileptic seizures using Riemannian geometry from scalp EEG recordings," *Medical & Biological Engineering & Computing*, vol. 59, pp. 1431-1445, 2021.

[2] H. Takahashi, A. Emami, T. Shinozaki, N. Kunii, T. Matsuo, and K. Kawai, "Convolutional neural network with autoencoder-assisted multiclass labelling for seizure detection based on scalp electroencephalography," *Computers in biology and medicine*, vol. 125, p. 104016, 2020.

[3] T. Prasanth, J. Thomas, R. Yuvaraj, J. Jing, S. S. Cash, R. Chaudhari, T. Y. Leng, R. Rathakrishnan, S. Rohit, V. Saini, and B. M. Westover, "Deep learning for interictal epileptiform spike detection from scalp EEG frequency sub bands," in *2020 42nd annual international conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020, pp. 3703-3706.

[4] M. Golmohammadi, V. Shah, I. Obeid, and J. Picone, "Deep learning approaches for automated seizure detection from scalp electroencephalograms," *Signal Processing in Medicine and Biology: Emerging Trends in Research and Applications*, pp. 235-276, 2020.

[5] X. Wang, W. Liu, Z. Chang, T. Kärkkäinen, and F. Cong, "One dimensional convolutional neural networks for seizure onset detection using long-term scalp and intracranial EEG," *Neurocomputing*, vol. 459, pp. 212-222, 2021.

[6] H. L. Chan, Y. Ouyang, P. J. Huang, H. T. Li, C. W. Chang, B. L. Chang, W. Y. Hsu, and T. Wu, "Deep neural networks for the detection of temporal-lobe epileptiform discharges from scalp electroencephalograms," *Biomedical Signal Processing and Control*, vol. 84, p. 104698, 2023.

[7] W. J. Chang, M. C. Chen, L. B. Chen, Y. C. Chiu, C. H. Hsu, Y. K. Ou, and Q. Chen, "A mobile device-based hairy scalp diagnosis system using deep learning techniques," in *2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech)*, Mar. 2020, pp. 145-146.

[8] X. Song, S. Guo, L. Han, L. Wang, W. Yang, G. Wang, and C. A. Baris, "Research on hair removal algorithm of dermatoscopic images based on maximum variance fuzzy clustering and optimization Criminisi algorithm," *Biomedical Signal Processing and Control*, vol. 78, p. 103967, 2022.

[9] R. M. Trüeb, "Oxidative stress and its impact on skin, scalp and hair," *International Journal of Cosmetic Science*, vol. 43, pp. S9-S13, 2021.

[10] M. Mosca, J. Hong, E. Hadeler, N. Brownstone, T. Bhutani, and W. Liao, "Scalp psoriasis: a literature review of effective therapies and updated recommendations for practical management," *Dermatology and Therapy*, vol. 11 no 3, pp.769-797, 2021.

[11] X. Hu, S. Yuan, F. Xu, Y. Leng, K. Yuan, and Q. Yuan, "Scalp EEG classification using deep Bi-LSTM network for seizure detection," *Computers in Biology and Medicine*, vol. 124, p.103919, 2020.

[12] Z. Fitzgerald, M. Morita-Sherman, O. Hogue, B. Joseph, M. K. Alvim, C. L. Yasuda, D. Vegh, D. Nair, R. Burgess, W. Bingaman, and I. Najm, "Improving the prediction of epilepsy surgery outcomes using basic scalp EEG findings," *Epilepsia*, vol. 62, no. 10, pp. 2439-2450, 2021.

[13] T. Dissanayake, T. Fernando, S. Denman, S. Sridharan, and C. Fookes, "Deep learning for patient-independent epileptic seizure prediction using scalp EEG signals," *IEEE Sensors Journal*, vol. 21, no. 7, pp. 9377-9388, 2021.

[14] H. Igaki, N. Murakami, S. Nakamura, N. Yamazaki, T. Kashihara, A. Takahashi, K. Namikawa, M. Takemori, H. Okamoto, K. Iijima, and T. Chiba, "Scalp angiosarcoma treated with linear accelerator-based boron neutron capture therapy: a report of two patients," *Clinical and Translational Radiation Oncology*, vol. 33, pp.128-133, 2022.

[15] B. Wei, J. Zhao, X. Shi, L. Xu, L. Liu, and J. Zhang, "A deep learning framework with multi-perspective fusion for interictal epileptiform discharges detection in scalp electroencephalogram," *Journal of Neural Engineering*, vol. 18, no. 4, p.0460b3, 2021.

[16] N. A. Enechukwu and A. O. Ogunbiyi, "A review of indigenous therapies for hair and scalp disorders in Nigeria," *Dermatologic Therapy*, vol. 35, no.6, p. e15505, 2022.

[17] Q. Wang, J. Ma, S. Yu, and L. Tan, "Noise detection and image denoising based on fractional calculus," *Chaos, Solitons & Fractals*, vol. 131, p. 109463, 2020.

[18] N. R Soora and P. S. Deshpande, "Review of feature extraction techniques for character recognition," *Journal of Research*, vol. 64, no. 2, pp. 280-295, 2018.

[19] S. Manimurugan, S. Al-Mutairi, M. M. Aborokbah, N. Chilamkurti, S Ganesan, and R. Patan, "Effective attack detection in internet of medical things smart environment using a deep belief neural network," *IEEE Access*, vol.8, pp. 77396-77404, 2020.

[20] R. Vatambeti, K. S Supriya, and S. Sanshi, "Identifying and detecting black hole and gray hole attack in MANET using gray wolf optimization," *International Journal of Communication Systems*, vol. 33, no. 18, p. e4610, 2020.

[21] Q. Man, L. Zhang, and Y. Cho, "Efficient hair damage detection using SEM images based on convolutional neural network," *Applied Sciences*, vol. 11, no. 16, p. 7333, 2021.

[22] W. J. Chang, L. B Chen, M. C. Chen, Y. C. Chiu, and J. Y. Lin, "ScalpEye: A deep learning-based scalp hair inspection and diagnosis system for scalp health," *IEEE Access*, vol. 8, pp. 134826-134837, 2020.

[23] A. Nabahin, A. Abou Eloun, and S. S. Abu-Naser, "Expert system for hair loss diagnosis and treatment," 2017.

[24] S. H Lee, and C. S. Yang, "An intelligent hair and scalp analysis system using camera sensors and Norwood-Hamilton model," *International Journal of Innovative Computing Information and Control*, vol. 14, no. 2, pp. 503-518, 2018

[25] W. C. Wang, L. B. Chen, and W. J. Chang, "Development and experimental evaluation of machine-learning techniques for an intelligent hairy scalp detection system," *Applied Sciences*, vol.8, no. 6, p. 853, 2018.

[26] J. H. Kim, S. Kwon, J. Fu, and J. H. Park, "Hair follicle classification and hair loss severity estimation using Mask R-CNN," *Journal of Imaging*, vol. 8, no. 10, p. 283, 2022.

[27] H. Kim, "Development of scalp diagnosis algorithm using surface-sensing convolutional neural network," *National Academy Science Letters*, pp.1-4, 2023.

[28] J. I. Jeong, D. S. Park, J. E. Koo, W. S. Song, D. J. Pae, and H. J. Choi, "Artificial intelligence (AI) based system for the diagnosis and classification of scalp health: AI-ScalpGrader," *Instrumentation Science & Technology*, vol. 51, no. 4, pp. 371-381, 2023.

[29] M. Kim, Y. Gil, Y. Kim, and J. Kim, "Deep-learning-based scalp image analysis using limited data," *Electronics*, vol. 12, no. 6, p. 1380, 2023.

[30] B. R. Kim, S. H. Won, J. W. Kim, M. Kim, J. I. Jeong, J. W. Shin, C. H. Huh, and J. I. Na, "Development of a new classification and scoring system for scalp conditions: Scalp Photographic Index (SPI)," *Journal of Dermatological Treatment*, vol. 34, no. 1, p. 2181655, 2023.

[31] M. Roy, and A. T. Protity, "Hair and scalp disease detection using machine learning and image processing," *arXiv preprint arXiv*, p.2301.00122, 2022.