Chinese Journal of Evidence-Based Medicine

Enhancing Thyroid Disease Detection through IWSO-based Ontology Matching and En-SwinT+ Classification Model

Rajeswari V	Hariharan R	Steffanio Rhynold R	Sasidharan S	Selvakumar M N
Professor	Final Year / CT	Final Year / CT	Final Year / CT	Final Year / CT
Department of CT	Department of CT	Department of CT	Department of CT	Department of CT
Karpagam college of Engineering,	Karpagam college of Engineering,	Karpagam college of Engineering,	Karpagam college of Engineering,	Karpagam college of Engineering,
Coimbatore,India	Coimbatore,India	Coimbatore,India	Coimbatore,India	Coimbatore,India
rajeswari.cst@gmail.com	hariiharan30@gmail.com	steffaniors07@gmail.com	sasisa66322@gmail.com	selvakumar6289@gmail.com

Abstract

Currently, medical informatics technology, Clinical decision support systems, and the medical technology are all making heavy use of medical ontologies. These coordinated and well-integrated systems are necessary for the smooth and accurate transmission of data and information. This, in turn, can lead to superior patient care and control of illnesses. Therefore, incorporating medical ontologies into these pathways is very important to its future success and development. Thyroid disease is a complicated health problem that has too much thyroid hormone. This is generally diagnosed with techniques such as CT scans and X-rays. The thyroid can be seen in the neck, just below the larynx. Thyroid problems can be found and solved by using Deep Learning (DL) technology in the proposed treatment. So, in this paper, to build semantic links between different datasets, this paper first apply Metaheuristic Optimization Strategy at Improved War Strategy Optimization (IWSO) at the beginning of our study to match ontologies properly. Subsequent to this, a preprocessing including Log and Gaussian filters is done to complete missing values and normalize features data for convenient comparison at the times of the modelling training. Then follows feature extraction, performed by the Gabor Wavelet Transform (GWT). For early detection of this disease, the study uses an Enhanced Swin transformer (En-SwinT+) model. This is a specially designed model by deep learning for just such diseases. Swin Transformer Inside has been refitted into a thyroid disease detection device. Our proposed model does better than the existing models. It attains an accuracy of 99. 45 % in the task of recognizing problems related to the thyroid gland. With the up-to-date and revolutionary methods employed in this research, thyroid disease identification effectiveness and accuracy can be greatly improved.

Keywords: Ontology, Thyroid disease detection, Gabor Wavelet Transform, Improved war strategy optimization, Swin transformer.

1. Introduction

When we are classifying this type then the systematic organization and the expression of medical thoughts and the collection and application of imaging data knowledge is the Huangong [1]. It uses a systematic approach to explain the links between illnesses, physical manifestations and their imaging features [2]. By relying on classification systems based on integrating ontological concepts, the differential diagnosis and treatment planning will be

more scientific and accessible. Ontologies help standardize medical terminologies and enable data from different sources of medical imaging to be processed [3]. Through this approach, different healthcare systems can exchange data, and data of poorly defined structure can be overcome, too [4]. Ontology, which can help collect the disorganized knowledge about how pictures are connected in detail, is a critical instrument for medical and image classification, and it might help spur on the computerized image interpretation Ford diagnosis. The thyroid gland's study of structure and function. There must, then, be a presence or absence of an anomaly in the structure or performance of that organ in order to diagnose any linked disease [5].

Hypothyroidism and hyperthyroidism affect metabolism. Timely action requires early detection. [6] Nuclear medicine and ultrasound are advanced imaging methods aiding in diagnosis. An algorithm trained on a large corpus of thyroid pictures better and more accurately determines abnormalities. This is what we mean by interdisciplinary science to make early, correct diagnoses of thyroid problems so that patients may be treated effectively [7]. Identifying thyroid illness is problematic due to symptoms' intricacy and early nature, variations in gland architecture from individual to individual, and challenges posed by imaging interpretation. Moreover, because benign and malignant instances overlap in features [8], it is hard to distinguish between them. These difficulties emphasize the need to develop sophisticated technology and complicated algorithms for accurate, timely identification [9].

Using multi-layer neural networks, deep learning can automatically extract hierarchical or multilevel representations from data. It is a prerequisite for classification [10]. Because it can recognize complex patterns and features, its superb accuracy allows voice or image recognition, natural language processing, and medical diagnosis [11]. Because deep learning models are so complex and advanced, they can extract valuable information from massive datasets. Their learning models include convolutional neural networks (CNNs) and recurrent neural networks (RNNs). This will make applications of artificial intelligence and machine learning more effective, making these techniques very efficient for complex classification in many fields [12,13].

1.1. Motivation

The emphasis placed by this trailblazing approach on the use of ontology is rooted in its powerful impact on advancing medical data integration and knowledge sharing, especially about detecting thyroid disease. The study creates a common standard in a structured ontology to systematically organize and present primary medical information by their respective relationships. This provides coherence in an otherwise complex field of thyroid diseases. This ontology is crucial in the powerful metaheuristic optimization technique, IWSO. It transforms medical images into relevant features for your analysis. More accurate and more efficient detection of thyroid disease not only improves the readability of diagnostic data but contributes to healthcare's more significant objective to get better patient results.

1.2. Main Contributions

- This paper emphasizes developing and using an ontology matching in thyroid disease detection. A novel metaheuristic optimization technique called Improved War Strategy Optimisation (IWSO) is applied to enhance the selection of significant features from

different datasets for thyroid image preprocessing, resulting in improved thyroid disease detection outcomes.

- Implements Log and Gaussian filters in the preprocessing step to refine and enhance critical features for further processing of thyroid images.
- This paper utilizes Gabor Wavelet Transform (GWT) to extract the features from preprocessed images.
- Adopts the En-SwinT^{+}, a specific type of Deep Learning model for classifying thyroid diseases. The Swin Transformer's internal architecture is optimized to ensure highly accurate detection.
- The proposed model demonstrates a significant advancement in accuracy, achieving an impressive 99.3% accuracy in detecting thyroid diseases.

2. Related Works

Alnaggar et al. [14] developed an excellent multiclass classification model using XGBoost. Sorting people according to their specific thyroid issues was their main priority. The two main advancements were a multiclass classification method that could differentiate between three different forms of thyroid diseases and enhanced feature selection accuracy. Notably, the very selective algorithm XGBoost demonstrated superior classification performance. Their model achieved an impressive 99% accuracy with hyperparameter optimization.

Prathibha et al. [15] established a paradigm for diagnosing thyroid problems using deep learning (DL) methods. To differentiate between hypergonadism, hypothyroidism, and thyroid nodules, among other thyroid disorders, they used a unique CNN-based ResNet architecture. Accuracy and efficiency were improved during training by using two optimizers. Their DL algorithms built with Keras within the Tensor Flow framework performed much better than previous studies in classifying thyroid diseases. The enhanced ResNet model attained an estimated 97% accuracy as opposed to the old design's 94% accuracy. In order to deduce thyroid problems from scanned photos, they also developed an online platform.

Kirsch's edge detector was used by Shankarlal et al. [16] to improve picture pixels in order to suggest a cancer detection technique. They used the Tree Contourlet Transform (DTCT) to obtain coefficients from the corrected pictures. These features were then used to train the Co-Active Adaptive Neuro segmentation method that could identify cancerous patches in thyroid imaging data. A CNN algorithm then established the tumour grade.

Brindha and Muthukumaravel's study [17] aimed to determine how accurately two classifier algorithms detected thyroid conditions. They compared the Support Vector Machine (SVM) and Convolutional Neural Network (CNN), using data from the UCI library to create models for training. In the case of detecting hypothyroidism and hyperthyroidism, CNN tides over SVM. On accuracy (89%) or precision (87%), respectively.

Wang et al. [18] designed an advanced learning-based computer-aided diagnostic (CAD) approach for predicting cervical lymph node metastasis (LNM) from CT scans of patients with thyroid cancer as their study objective. Their system fuses CT data analysis and

advanced LN identification by applying region-based detection networks and residual networks with attention modules.

Shankarlal et al. [19] introduced a computer-aided method for identifying tumours and their boundaries in thyroid ultrasound images using machine learning and deep learning techniques. Their technique used the co-active Adaptive Neuro Expert System (CANFES) classifier for feature computation, edge augmentation, transform application and classification. After the tumours were located by morphological segmentation, they used CNN to classify the extra diagnosis.

Zhang et al. [20] classified thyroid disease groups using ultrasounds and computed tomography (CT) images. Using a cutting-edge deep convolutional neural network architecture (CNN), they created a diagnostic model to distinguish between different thyroid conditions. Their model suggests that visual modalities have good accuracy for both ultrasound (0.972) and CT scans (0.942); they may be utilized to identify thyroid problems.

2.1. Research gap

Related work has made rapid progress using machine learning and deep learning methods for diagnosing thyroid disease with imaging data. However, a gap between these advanced techniques and medical ontologies exists, leading to less complete diagnostic support systems. However, these deep learning models produce impressive accuracy rates ranging from 97 % to 99.45 %, using imaging tools like CT scans and ultrasounds, so there is a need for a bridge between them and medical ontologies. Implementing ontologies in diagnostic systems could lead to more easily understood results, a better understanding of the problems detected, and standardized terminology for thyroid disorders. A methodology capable of integrating these deep learning models with medical ontologies could help tremendously in better clinical decision-making. With this, patients will receive more effective and informed treatment strategies for thyroid disease.

3. Proposed Methodology

This study proposes an ontology-based classification system for thyroid diseases. The steps of the suggested approach are shown in Figure 1. The initial step is the ontology matching using IWSO, image preprocessing with Log and Gaussian filter, feature extraction using GWT and enhanced Swin Transformer classification.

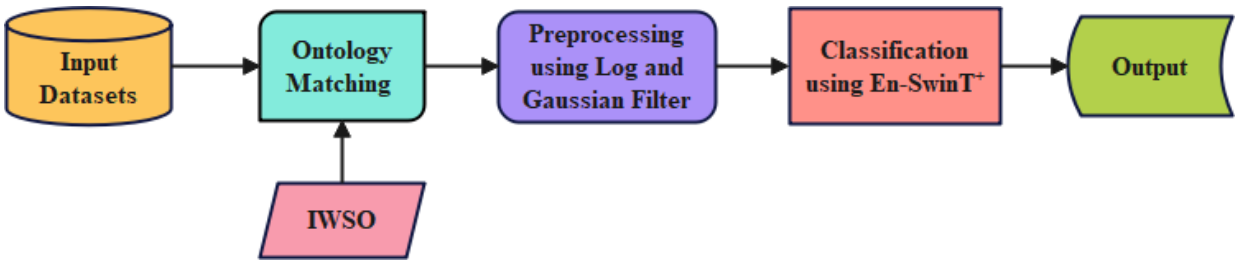


Figure 1. Workflow of the proposed model

Vol. 20, No. 1. (2024)
E ISSN: 1672-2531

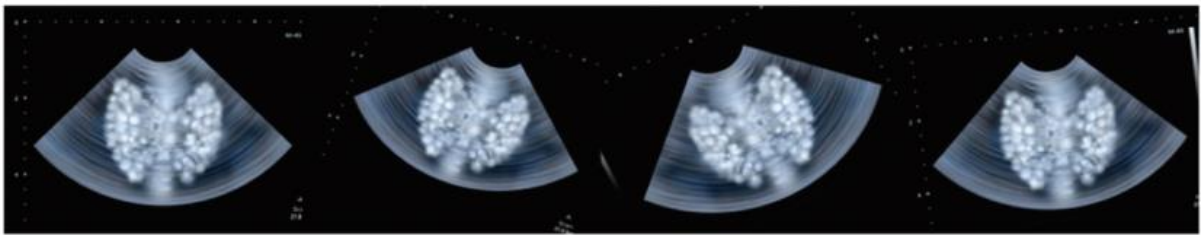
3.1. Dataset Collection

Most earlier research has evaluated deep learning or machine learning models using CT or X-ray scan datasets from nearby hospitals. Data is so valuable in the deep learning era that many startups have offered image annotation services to collect and label data. Kaggle is one well-known platform to evaluate the suggested model [21–25]. New thyroid datasets from the UCI repository, including ultrasound images for hyper- and hypothyroid diseases, have also been utilized to evaluate the suggested system's performance. The quantity of datasets collected for the assessment of the proposed method is shown in Table 1:

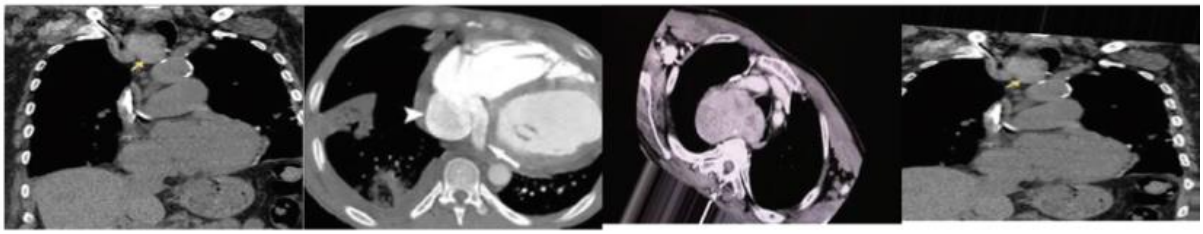
Table 1: Dataset collection

Type of Disease	No. of images
Thyroid cancer	99
Hyperthyroid	77
Hypo thyroid	110
Thyroid nodules	146
Normal thyroid	88
Thyroiditis	74

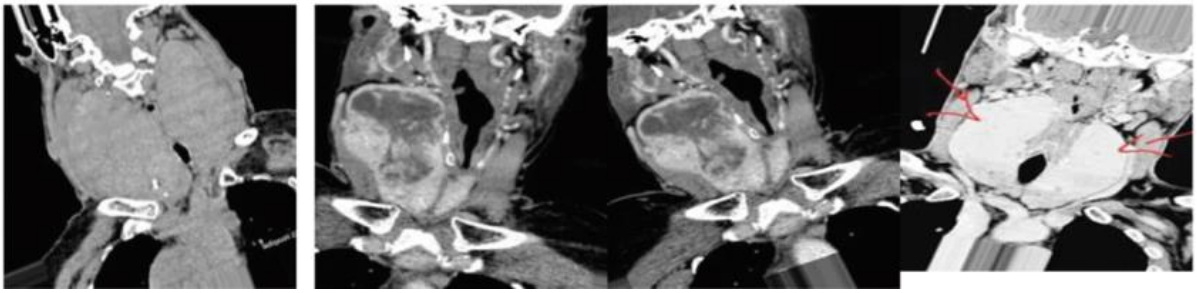
The gathered datasets have been classified according to the diseases they represent. Figure 2 shows a sample image of each disease.



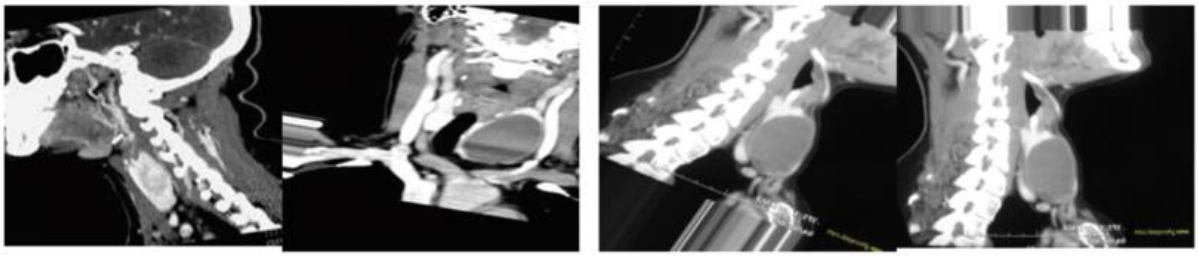
(a) Thyroid Cancer



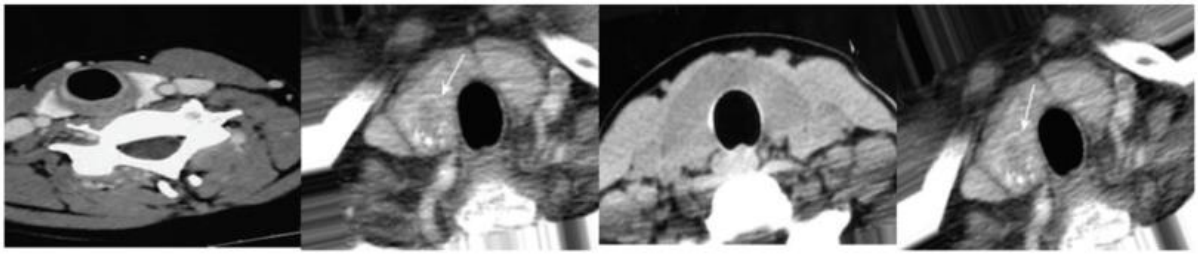
(b) Hyperthyroid



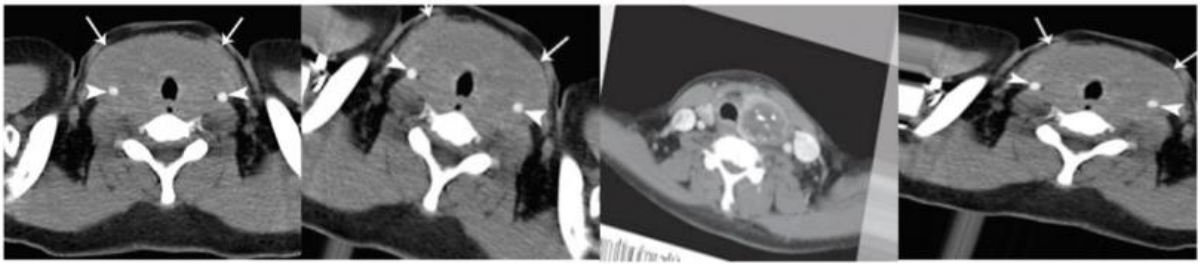
(c) Hypothyroid



(d) Thyroid Nodules



(e) Normal Thyroid



(f) Thyroiditis

Figure 2. Sample datasets

3.2. Ontology Matching using IWSO

This paper uses Improved war strategy optimization for ontology matching between two different datasets.

3.2.1. Basic Concepts

Definition 1 (Ontology): An ontology O is a tuple: $O = (\mathbb{C}, \mathbb{R}, \mathbb{Q}, V)$

- C is a collection of classes that Owl has created: Class.
- $\mathcal{R}$ is a collection of relations that characterize the relationship between objects.
- Q symbolizes a collection of examples within a specific domain.
- V symbolizes a collection of annotations used to give entities descriptive information.

Vol. 20, No. 1. (2024)
E ISSN: 1672-2531

Definition 2 (Ontology matching): The process by which an algorithm discovers a set of correspondences between entities that belong to different ontologies is known as ontology matching. The equation (1) can be used to formalize it:

$$A = f(O_1, O_2, \zeta, \Gamma) \quad (1)$$

Where A is an alignment, or corresponding result, between O_1 and O_2, ζ is a collection of materials that includes more straightforward matches or background information, and Γ is a collection of parameters used in ontology matching.

Definition 3 (Entity): Within the ontology framework, an entity denotes a specific class or property; for example, a class entity represents a class, and a property entity represents a property.

Definition 4 (Correspondence). A mapping or correspondence between an ontology-derived entity e_1, O_1 and an entity e_2 coming from ontology O_2 is in this work defined as a 5-tuple:

$$\text{Cor} = \langle e_1, e_2, r, \eta, g \rangle$$

Where:

- r is a connection between the things e_1 and e_2, like equivalency ($\equiv$) and broader ($\sqsupseteq$).
- η is an entity's similarity value e_1 and e_2.
- g is a mark that has been created especially for the suggested strategy.

The methodology recommends incorporating the new IWSO metaheuristic optimization technique to support its ontology matching strategy. Customized to build ontologies, this approach dramatically affects the choice of critical characteristics needed for additional thyroid image processing. Using IWSO, the system can identify and highlight important features in thyroid images, improving the effectiveness of the image analysis procedures.

3.2.2. War Strategy Optimization Algorithm

The WSO algorithm was first presented by Ayyarao et al. in 2022 [26], and historical military strategies inspired it. Applying soldier movements to solve optimization problems imitates methods used in previous battles. During the search process of the WSO algorithm, all possible solutions come together to form an army, with each solution serving as a soldier. The king commands the military, with the strongest soldier serving as commander. The soldiers constantly reposition themselves as the commander and king move. Considering the current status of the "war," the king plans a methodical offensive.

There are regular status updates for every soldier, including recruitment or relocation tactics in case of injuries or unfavourable placements. The WSO algorithm uses four update strategies to determine the position of soldiers. However, its current method produces slow convergence and low accuracy. This problem is addressed by the improved WSO algorithm, which incorporates dynamic step sizes, improves result accuracy, and speeds up problem-solving. This is a summary of the improved WSO algorithm's mathematical representation:

- Attack strategy

To update their positions, soldiers strategically rearrange themselves about the positions of the commander and king. Equation (2) provides a mathematical description of this process:

$$X_i(t+1) = X_i(t) + 2 \cdot \text{rand} \cdot (C - \text{King}) + \text{rand} \cdot \left(W_i \cdot \text{King} - X_i(t)\right) \quad (2)$$

The soldier's new position at iteration $X_i(t+1)$ is represented by $t+1$, $X_i(t)$, and the soldier's position at iteration t is indicated by $X_i(t)$. Here, the terms " " and "King" denote the king and the commander in that order, and "W_i" indicates the importance of the king's status.

- Rank and weight updating

Soldiers choose to be in better positions to increase their advantage in combat, directly affecting their combat efficiency and rank. The following is represented mathematically in equations (3) and (4):

$$X_i(t+1) = (X_i(t+1)) \cdot \left(F_n \geq F_p\right) + (X_i(t)) \cdot \left(F_n < F_p\right) \quad (3)$$

$$R_i = (R_i + 1) \cdot \left(F_n \geq F_p\right) + (R_i) \cdot \left(F_n < F_p\right) \quad (4)$$

F_n denotes the Combat effectiveness of the soldier in the new role, F_p shows how successful they were in their prior position, and R_i represents the soldier's rank.

Soldiers are ranked according to their effectiveness, which affects the modified weight, as shown by equation (5).

$$W_i = W_i \cdot (1 - R_i/T)^{\alpha} \quad (5)$$

Here, α denotes an exponential factor, and T represents the number of iterations.

- Defence strategy

Soldiers measure their distance from the king in times of war to make sure his safety is the top priority, as shown in equation (6):

$$X_i(t+1) = X_i(t) + 2 \cdot \text{rand} \cdot (\text{King} - X_{\text{rand}}(t)) + \text{rand} \cdot W_i \cdot (C - X_i(t)) \quad (6)$$

The symbol for the soldier's random position at iteration t is $X_{\text{rand}}(t)$.

- Replacement/Relocation of weak soldiers

To improve the standing of soldiers who are not as strong, the WSO algorithm uses two different strategies. First, using equation (7) as a guide, it moves the weaker soldiers to the middle of the battlefield. The second way it uses equation (8) is to create new soldiers randomly.

$$X_w(t+1) = Lb + \text{rand} \cdot (Ub - Lb) \quad (7)$$

$$X_w(t+1) = -(1 - \text{randn}) \cdot (X_w(t) - \text{median}(X)) + \text{King} \quad (8)$$

$X_w(t+1)$ denotes the iteration in place of the feeble soldiers $t+1$, Ub, and Lb for the respective upper and lower value limits. A uniformly distributed random number between 0 and 1 is denoted by the symbol *random; the median function is represented by the symbol median (·).

3.2.3. IWSO Algorithm

An upgrade was made to the WSO algorithm to improve the approach to changing ranks and weights and reorganizing the weaker soldier replacement or relocation procedure. These changes were made to improve the algorithm's overall performance.

- **Better updating of ranks and weight**

Soldier ranks were initially calculated by the algorithm using a fixed exponential factor. A dynamic approach was used to improve both the early-stage global search and late-stage convergence capabilities. This required modifying equation (9)'s exponential factor according to the iteration number (t).

$$\alpha = 2\left(1 - e^{-t/T}\right) \quad (9)$$

- **Better substitution or transfer of less capable soldiers**

Initially, the algorithm used a fixed exponential factor to determine soldier ranks. A dynamic approach was used to improve both the early-stage global search and late-stage convergence capabilities. This required modifying equation (10)'s exponential factor according to the iteration number (t).

$$X_w(t+1) = \begin{cases} \text{Lb r rand} \cdot (Ub - Lb) & t \le \frac{1}{3}T \\ \beta(Lb + \text{rand} \cdot (Ub - Lb)) + (1-\beta)(-(1 - \text{randn}) \cdot (X_w(t) - \text{median}(X)) + \text{King}) & \frac{1}{3}T < t \le \frac{2}{3}T \\ -(1 - \text{randn}) \cdot (X_w(t) - \text{median}(X)) + \text{King} & t > \frac{2}{3}T \end{cases} \quad (10)$$

$$\beta = e^{1-\frac{3t}{T}} \quad (11)$$

Moreover, Figure 3 displays the flow chart of the IWSO algorithm. All these changes are aimed at improving the algorithm's performance and efficiency.

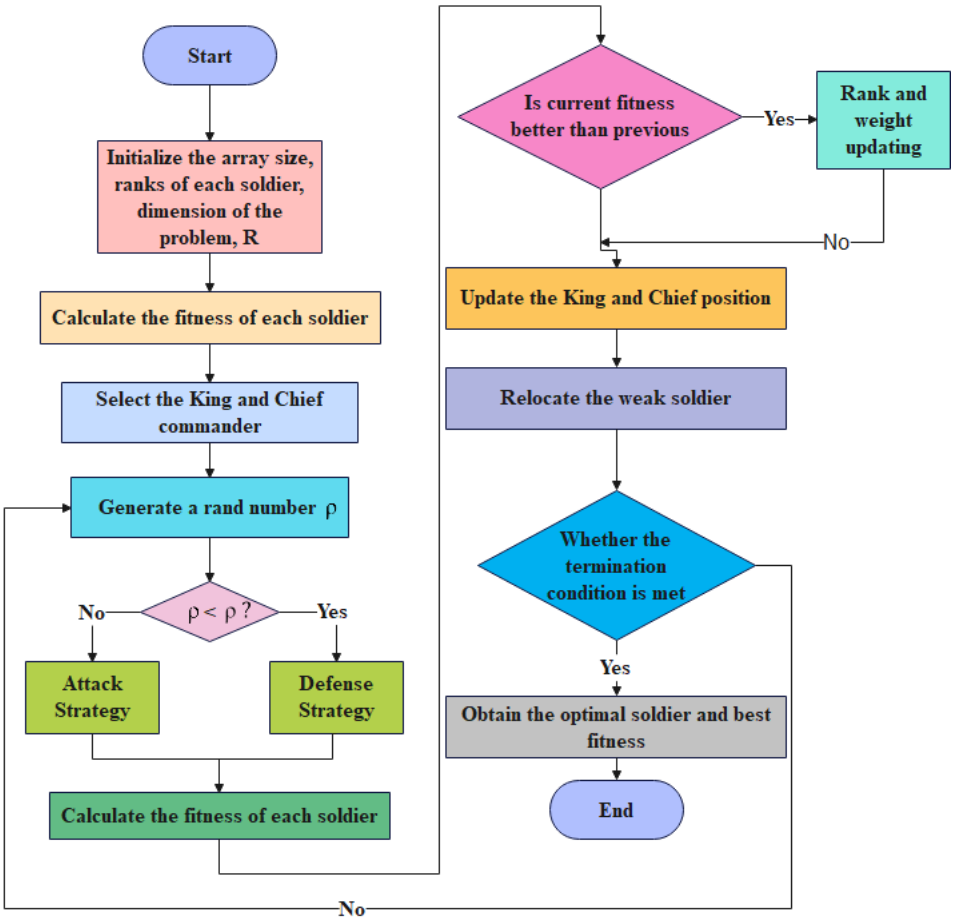


Figure 3. Flowchart of IWSO

3.3. Preprocessing using Log and Gaussian Filters

After important features are extracted using IWSO from a thyroid image dataset, these features are preprocessed. Using a log filter reduces noise and aids in edge recognition in thyroid images. This study applied a log filter to both the reference and distorted thyroid images, and then a Gaussian filter [27] was used to reduce noise further. Equation (12), where σ stands for the standard deviation and $L(p,q)$ indicates the location-based (p,q) filter, is the mathematical expression for the Log filter.

$$L(p,q) = -\frac{1}{\pi\sigma^4}\left(1-\frac{p^2+q^2}{2\sigma^2}\right)\exp\left(-\frac{p^2+q^2}{2\sigma^2}\right) \quad (12)$$

3.4. GWT Feature Extraction

A helpful texture description can be obtained using GWT to extract multi-scale features from ultrasound images at different orientations [28]. The Gaussian kernel feature of the Gabor filter is used to process a complex sinusoidal signal on its spatial region. Equation (13) represents the Gabor filter's sinusoidal signal.

$$g = \frac{s^2}{\pi\gamma\eta}\exp\left(-\frac{x'+\gamma^2 y'}{2\sigma^2}\right)\exp\left(j2\pi s x'+\varphi\right) \quad (13)$$

Here, θ is the frequency's orientation in phase offset φ. The standard deviation σ is applied to the Gaussian envelope, and the elliptic features are shown by γ. Equations (14) and (15) below delineate the x' and y' sections.

$$x' = x\cos\ \theta + y\sin\ \theta \quad (14)$$

$$y' = x\sin\ \theta + y\cos\ \theta \quad (15)$$

Six scales and four orientations are used to create 24 multi-scale images for the GWT. For these multidirectional images, four statistical features are measured: entropy, energy, homogeneity, and correlation.

3.5. En-SwinT$^+$ model

The En-SwinT$^+$ model makes classification easier after post-preprocessing. Its Patch Encoder uses a convolutional layer for feature extraction and a Patch Partitioner to create patches from the previously processed image. The Swin Transformer that is being suggested $^+$ different approaches to patch partitioning and embedding between the model and the Swin Transformer model [29].

Regarding the Swin Transformer prototype, patch sizes are tailored to contain individual dynamic objects, leading the preprocessed image p_t to be divided into a set of fixed-size patches $p_{t,1}, p_{t,2}, \ldots, p_{t,n}$ by the Patch Partitioner and n denotes the number of pixels. The token features $c_{t,1}, c_{t,2}, \ldots, c_{t,n}$ are obtained from the raw RGB pixel values using a convolutional layer, as shown in Figure 4 and explained in equation (16).

$$c_{t,n} = \mathrm{Conv}\left(f^p(p_t)\right)(n = 1,2, \ldots, N) \quad (16)$$

Conv, f^p, represents the convolutional layer $f^p(\cdot)$, which signifies Patch Partitioner, and N represents the total number of patches.

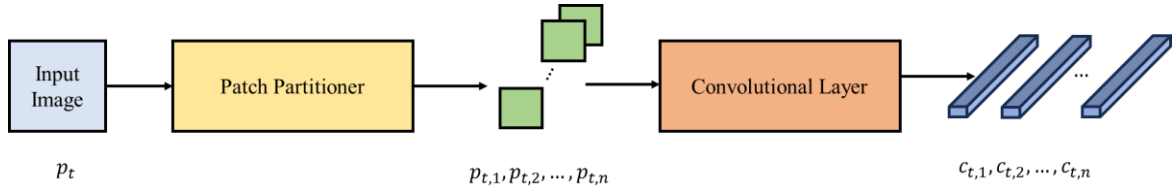


Figure 4. The Patch Encoder Process.

The Swin Transformer $^+$ Blocks (designated as (a), (b), and (c)) that are in charge of feature extraction and augmentation are housed in three stages of the Swin Transformer $^+$ model. These blocks, which are made up of a Window Attention Processor, MLP, and Padding Handler, receive token sets $C_t = \{c_{t,1}, c_{t,2}, \ldots, c_{t,n}\}$ as input and produce features that are used for the prediction of subsequent actions, taking associated losses into account.

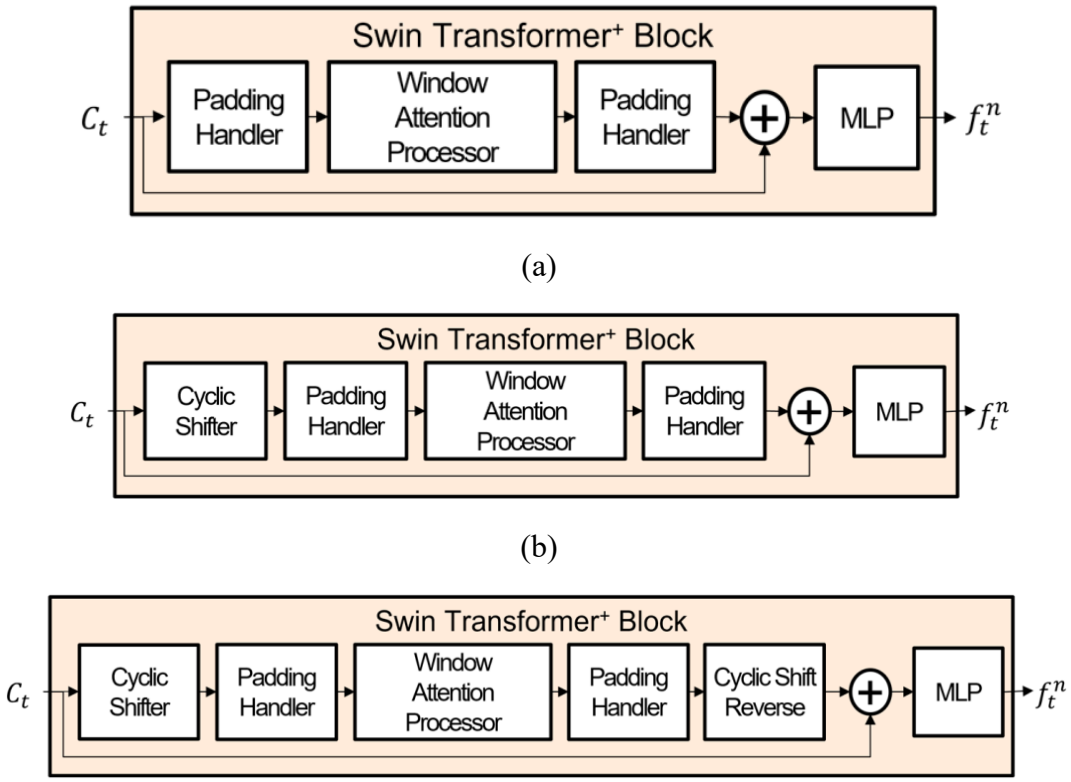


Figure 5. The Swin Transformer $^+$ Block configuration, arranged from above (a-c): a) The initial Swin Transformer Block in the stage; b) The subsequent Swin Transformer $^+$ Block in the stage; c) The final Swin Transformer $^+$ Block in the stage.

Because the Swin Transformer $^+$ Blocks rely on a window size as a basis for token grouping, they differ from the Swin Transformer $^+$ Blocks in terms of quantity and structure. Moreover, a shift by $\left\lfloor \frac{W}{2} \right\rfloor$ and a window of size W are needed to rearrange tokens. This requires a change by $\left\lceil \frac{W}{2} \right\rceil$. Therefore, the Swin Transformer $^+$ Blocks equals $\left\lceil \frac{W}{2} \right\rceil + 1$ plus one more step for the original token arrangement.

Every Swin Transformer $^+$ Block is built differently. A Padding Handler is incorporated into each of these blocks before and after the Window Focus Processor to support dynamic elements within fixed patch sizes. The Swin Transformer Block does not include this particular function. For example, Figure 5a shows that the Padding Handler is only integrated into the Swin Transformer Block by the first Swin Transformer $^+$ Block. Figure 5b, however, shows the second Swin Transformer $^+$ Block positioned in the centre, which improves features by applying successive cyclic shifts inside the token group. Nevertheless, the third Swin Transformer $^+$ Block is the only one where token order restoration occurs (Figure 5c). Consequently, the sequence within Swin Transformer $^+$ Blocks follows a periodic change contrary to the Swin Transformer's cyclic shift-reverse-cyclic shift-reverse pattern, which is cyclic.

The Padding Handler, a crucial part of the Swin Transformer+ Block, applies zero-value padding to guarantee consistent token distribution about window size. If the height or width of the tokens does not match the size of the window, then more padding is added to the rows or columns. However, these additional paddings help with token grouping within windows but do not affect attention calculations. As a result, the Window Attention Processor comes before and after the Padding Handler, which runs twice: once for padding addition and once for removal following self-attention. Equation (17) represents the mechanism that the Padding Handler uses:

$$\begin{aligned} t^{w'} &= t^{w} + n, n = W - (t^{w} \% W) \\ t^{h'} &= t^{h} + n,\ n = W - (t^{h} \% W) \end{aligned} \quad (17)$$

The values tw and th in the given equation represent the token count in width and height. When dividing t^w and t^h by W, the number of deficiencies is indicated by the symbol n, which represents the window size. The symbol % shows the remainder after dividing a number. When t^w and t^h are not evenly divisible by W, additional padding is added by an amount equal to n. To calculate n, divide both $t^w and\ t^h$ by W, then deduct the remaining amount from W.

Regarding data augmentation, The Cyclic Shifter initiates a cyclic shift at the stage's second Swin Transformer $^+$ Block. This change keeps each token's encoded RGB pixels intact but changes the order in which they appear. Popping tokens around is a form of data augmentation since each token represents a dynamic object.

The Attention Processor Window uses padding and cyclic shift masks to apply self-attention inside a window. By doing this, paddings and pointless tokens are kept from impacting the attention mechanism. In the n-th window, the k-th token's value, for example, is indicated by e_k^n, and the cyclic shift mask's value is displayed by $c_{k'}^n$. The dimensions of the padding mask are $W \times W$, and it is generated according to the window size W. To ensure that padded areas do not impact the attention score computation, values in the padding mask (d_k^n) are modified.

The padding masks guarantee that attention scores are not affected by additional paddings. They function by summing values using the cyclic shift mask, keeping the unaffected areas of the show intact while filtering out high-magnitude padding mask values through softmax to almost zero.

The operation of these masks is shown in Figure 5. Yellow rectangles represent tokens, grey rectangles represent additional padding, and the window's outline is black. The padding mask's black and white colouring represents negative padding values and zero token values, respectively.

Vol. 20, No. 1. (2024)
E ISSN: 1672-2531

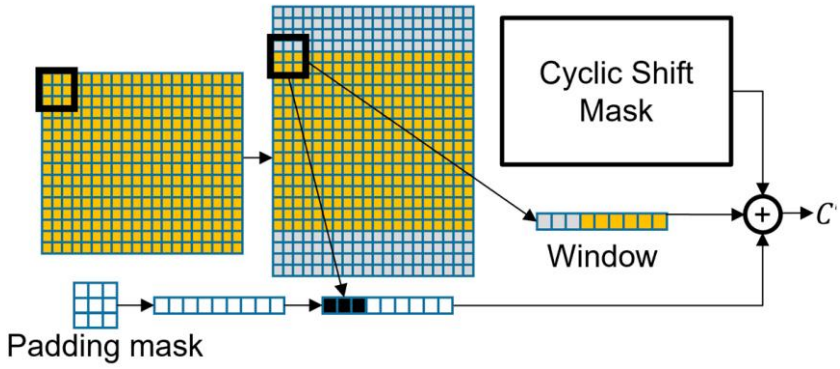


Figure 6. The Window Attention Processor.

As a result, the final result, defined as C'_t, combines the attention outcomes from each of the image's windows [30].

$$d_k^n = \begin{cases} -100 & \text{if } e_k^n \text{ is padding} \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

$$C_t' = \sigma(WAP(e_k^n) + d_k^n + c_k^n)(k = 1,2,\dots,W^2) \quad (19)$$

Equations (18) and (19) are used by the Window Attention Processor C'_t to generate its output. In this case, c_k^n denotes the cyclic shift mask's value, $\sigma(\cdot)$ is the softmax function, and WAP stands for Window Attention Processor.

The MLP functions similarly to the Swin Transformer in that it blends features. The Swin Transformer is combined with the output feature $^+$ Block's input through a residual connection, and the MLP is used to connect the two further. Because there are n blocks and i stages, each Swin Transformer $^+$ Block extracts features, which results in $n \times i$ parts.

4. Results and Discussion

4.1. Experimental Setup

The studies were conducted on Google Colab, a cloud computing platform that provides GPU and TPU resources at no cost. The process moved along much faster when training on a GPU instead of a PC. Using a Tensorflow GPU backend and the Python Keras library, the proposed model was applied to the dataset's image classification. Table 2 contains comprehensive information about the hardware and software used in these experiments.

Table 2: Specifications for the experiment's hardware and software.

Hardware	Software
Processor: core i5 2.2 gigahertz	Programming language: Python version 3.9
RAM: 32 gigabytes	OWLReady: Under the GNU LGPL licence v3

Graphical Processing Unit (GPU)	Backend: Tensorflow GPU
Hard drive: 500 gigabytes NVIDIA, 16 gigabytes RAM	Deep learning API: Keras GPU

4.2. Performance Metrics

True positive (TP), true negative (TN), false positive (FP), and false negative (FN) are the four primary analytical metrics that this study computed to evaluate the effectiveness of the classification system built using the datasets.

When evaluating the efficacy of a classification model, ACC is described as the ratio of correct assumptions to total assumptions made in equation (20):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (20)$$

PR, also known as positive predictive value in equation (21), measures the ratio of all positive examples to the number of correctly identified positive models:

$$Precision = \frac{TP}{TP+FP} \quad (21)$$

RC called the sensitivity or actual positive rate, represents the proportion of adequately determined positive cases out of all positive instances, as shown in equation (22).

$$Recall = \frac{TP}{TP+FP} \quad (22)$$

The F1 in equation (23) is an integrated metric that incorporates PR and RC into a single numerical value:

$$F1 = \frac{Precision*Recall}{Precision+Recall} \quad (23)$$

4.3. Performance Validation

Table 1: Various classes of thyroid disease detection

Classes	ACC	PR	RC	F1
Thyroid cancer	97.52	97.72	97.25	97.41
Hyperthyroid	97.41	97.14	97.63	97.22
Hypo thyroid	98.54	98.43	97.32	98.33
Thyroid nodules	98.43	97.71	98.61	98.54
Normal thyroid	98.56	98.92	97.96	97.92
Thyroiditis	99.45	98.93	98.84	98.82

Table 1 and Figure 7 present performance metrics for the detection of various classes of thyroid diseases, showcasing the accuracy (ACC), precision (PR), recall (RC), and F1 score for each category. Thyroid cancer detection achieved an accuracy of 97.52%, with a precision of 97.72%, recall of 97.25% and F1-Score of 97.41%, respectively. Hyperthyroidism detection exhibited an accuracy of 97.41%, with precision, recall, and F1 score values of 97.14%, 97.63%, and 97.22%. A high accuracy of 98.54% was shown in detecting hypothyroidism, with corresponding precision, recall, and F1 score values of

98.43%, 97.32%, and 98.33%. 98.43% of thyroid nodules were found, with related precision, recall, and F1 score values of 97.71%, 98.61%, and 98.54%. The accuracy of the standard thyroid classification was 98.56%, with precision, recall, and F1 scores of 98.92%, 97.96%, and 97.92%. Finally, thyroiditis detection showed exceptional performance with an accuracy of 99.45%, precision, recall, and F1 score values of 98.93%, 98.84%, and 98.82%, respectively. These results highlight the proposed thyroid disease detection system's remarkable accuracy and precision in differentiating six classes. The evaluation proved that the proposed model is effective across multiple classes.

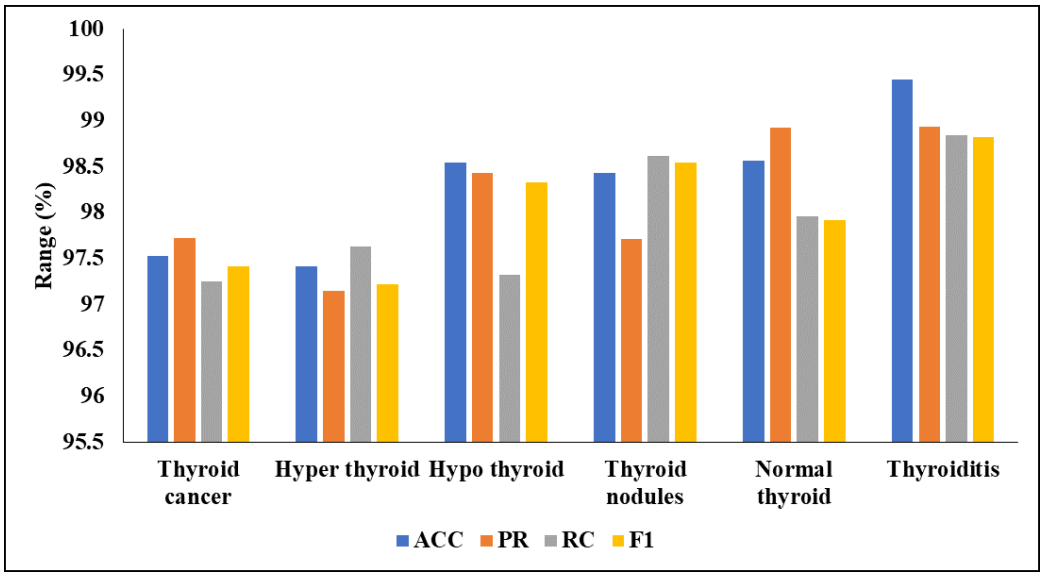


Figure 7. Evaluation of Various classes

Table 2: Comparing the proposed model to alternative models.

Models	ACC	PR	RC	F1
LeNet	95.32	95.13	94.33	95.02
AlexNet	96.42	96.24	95.85	96.13
VGGNet	97.34	97.35	96.97	97.34
CapsNet	98.13	98.27	97.20	98.46
Proposed En-SwinT^{+} Model	99.45	99.22	98.96	99.07

A detailed comparison of the proposed En-SwinT^{+} Model across many forms of accuracy measures can be found in Table 2 and Figure 8, along with comparisons to several other well-adopted models such as LeNet, AlexNet, VGGNet and CapsNets. The assessment criteria are F1 score, recall (RC), accuracy (ACC) and precision (PR). Using an astonishingly accurate En-SwinT^{+} Model (99.45 %), it not only beats all the other models but demonstrates its superior classification abilities. However, the model is also excellent in accuracy (99.22%), recall (98.96%) and F1 score (99.07%). Compared with traditional and widely used models such as LeNet, AlexNet, VGGNet and even Caps- Net (the novelty of which is

noteworthy), the proposed En-SwinT^{+} Model attainably holds the best score for every metric. This verifies its ability to tackle the thyroid disease classification.

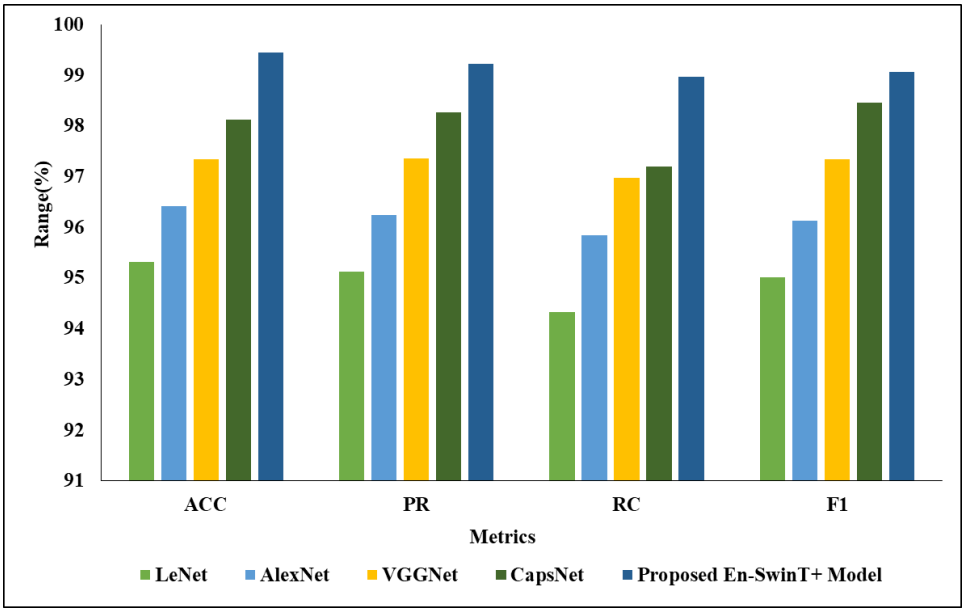


Figure 8. Comparison of the proposed model with existing models

Table 3: Existing model comparison

Models	Accuracy (%)
XGBoost [14]	99
Modified ResNet [15]	94
CAD [18]	96
CANFES [19]	99.15
Deep CNN [20]	97.2
Proposed En-SwinT^{+} Model	99.45

Experiments to compare the accuracy of different models, Table 3 and Figure 9 evaluate each model's performance based on metrics. The model with the highest accuracy of 99.45 % is found among other models assessed, namely the Proposed En-SwinT^{+} Model. The results of this model far outperform those from other noted methods (XGBoost 99 %; CANFES 98.15%; Modified ResNet 94%; and Deep CNN 97.2%). Looking at this comparison, it is clear that the Proposed En-SwinT^{+} Model outperforms both modified and traditional versions for achieving accurate thyroid disease classification.

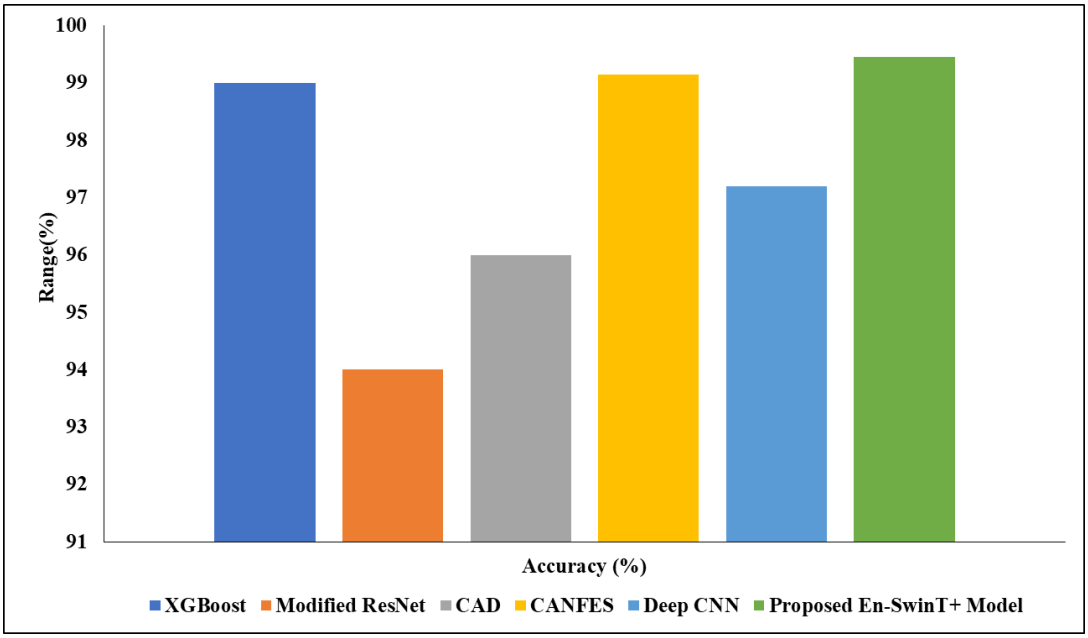


Figure 9. Comparison of existing models with the proposed model.

5. Conclusion

Finally, integrating medical ontologies, IWSO, for ontology matching in thyroid datasets is a critical link between medical informatics, clinical decision support systems, and the cutting-edge technologies used to diagnose thyroid disease. Preprocessing, such as applying Log and Gaussian filters, can improve data quality. For feature extraction, Gabor Wavelet Transform proved effective. Incorporating the Enhanced Swin Transformer (En-SwinT+) model highlights how tailored deep learning solutions can work, with a remarkable 99.45 % accuracy in identifying thyroid disease. The research also emphasizes the need for ontology matching in medical pathways and provides a systematic, advanced technical method with high potential to improve the accuracy of thyroid disease identification. The methodology put forward could be further extended to various medical fields. Its effectiveness beyond the field of thyroid disease needs to be explored in future work. More importantly, trying out real-time applications and integrating the model into clinical environments could enhance its practicality and expandability.

References

[1]. Elhadj, H. B., Sallabi, F., Henaien, A., Chaari, L., Shuaib, K., & Al Thawadi, M. (2021). Do-Care: A dynamic ontology reasoning based healthcare monitoring system. *Future Generation Computer Systems*, *118*, 417-431.

[2]. Subramaniyaswamy, V., Manogaran, G., Logesh, R., Vijayakumar, V., Chilamkurti, N., Malathi, D., & Senthilselvan, N. (2019). An ontology-driven personalized food recommendation in IoT-based healthcare system. *The Journal of Supercomputing*, *75*, 3184-3216.

[3]. Chen, L., Lu, D., Zhu, M., Muzammal, M., Samuel, O. W., Huang, G., ... & Wu, H. (2019). OMDP: An ontology-based model for diagnosis and treatment of diabetes patients in remote healthcare systems. *International Journal of Distributed Sensor Networks*, *15*(5), 1550147719847112.

[4]. Luschi, A., Petraccone, C., Fico, G., Pecchia, L., & Iadanza, E. (2023). Semantic ontologies for complex healthcare structures: a scoping review. *IEEE Access*.

[5]. Chaubey, G., Bisen, D., Arjaria, S., & Yadav, V. (2021). Thyroid disease prediction using machine learning approaches. *National Academy Science Letters*, *44*(3), 233-238.

[6]. Sonuç, E. (2021, July). Thyroid disease classification using machine learning algorithms. In *Journal of Physics: Conference Series* (Vol. 1963, No. 1, p. 012140). IOP Publishing.

[7]. Alyas, T., Hamid, M., Alissa, K., Faiz, T., Tabassum, N., & Ahmad, A. (2022). Empirical method for thyroid disease classification using a machine learning approach. *BioMed Research International*, *2022*.

[8]. Abbad Ur Rehman, H., Lin, C. Y., Mushtaq, Z., & Su, S. F. (2021). Performance analysis of machine learning algorithms for thyroid disease. *Arabian Journal for Science and Engineering*, 1-13.

[9]. Aversano, L., Bernardi, M. L., Cimitile, M., Iammarino, M., Macchia, P. E., Nettore, I. C., & Verdone, C. (2021). Thyroid disease treatment prediction with machine learning approaches. *Procedia Computer Science*, *192*, 1031-1040.

[10]. Raisinghani, S., Shamdasani, R., Motwani, M., Bahreja, A., & Raghavan Nair Lalitha, P. (2019). Thyroid prediction using machine learning techniques. In *Advances in Computing and Data Sciences: Third International Conference, ICACDS 2019, Ghaziabad, India, April 12–13, 2019, Revised Selected Papers, Part I 3* (pp. 140-150). Springer Singapore.

[11]. Prathibha, S., Dahiya, D., Robin, C. R., Nishkala, C. V., & Swedha, S. (2023). A Novel Technique for Detecting Various Thyroid Diseases Using Deep Learning. *Intell. Autom. Soft Comput*, *35*(1), 199-214.

[12]. Mir, Y. I., & Mittal, S. (2020). Thyroid disease prediction using hybrid machine learning techniques: An effective framework. *International Journal of Scientific & Technology Research*, *9*(2), 2868-2874.

[13]. Guleria, K., Sharma, S., Kumar, S., & Tiwari, S. (2022). Early prediction of hypothyroidism and multiclass classification using predictive machine learning and deep learning. *Measurement: Sensors*, *24*, 100482.

[14]. Alnaggar, M., Handosa, M., Medhat, T., & Rashad, M. Z. (2023). Thyroid disease multi-class classification based on optimized gradient boosting model. *Egyptian Journal of Artificial Intelligence*, *2*(1), 1-14.

[15]. Prathibha, S., Dahiya, D., Robin, C. R., Nishkala, C. V., & Swedha, S. (2023). A Novel Technique for Detecting Various Thyroid Diseases Using Deep Learning. *Intell. Autom. Soft Comput*, *35*(1), 199-214.

[16]. Shankarlal, B., Sathya, P. D., & Sakthivel, V. P. (2023). Computer-aided detection and diagnosis of thyroid nodules using machine and deep learning classification algorithms. *IETE Journal of Research*, *69*(2), 995-1006.

[17]. Brindha, V. and Muthukumaravel, A., 2023. Efficient Method for Predicting Thyroid Disease Classification using Convolutional Neural Network with Support Vector Machine. In *Computational Intelligence for Clinical Diagnosis* (pp. 77-85). Cham: Springer International Publishing.

[18]. Wang, T., Yan, D., Liu, Z., Xiao, L., Liang, C., Xin, H., Feng, M., Zhao, Z. and Wang, Y., 2023. Diagnosis of cervical lymph node metastasis with thyroid carcinoma by deep learning application to CT images. *Frontiers in Oncology*, *13*, p.1099104

[19]. Shankarlal, B., Sathya, P. D., & Sakthivel, V. P. (2023). Computer-aided detection and diagnosis of thyroid nodules using machine and deep learning classification algorithms. *IETE Journal of Research*, *69*(2), 995-1006.

[20]. Zhang, X., Lee, V.C., Rong, J., Lee, J.C. and Liu, F., 2022. Deep convolutional neural networks in thyroid disease detection: a multi-classification comparison by ultrasonography and computed tomography. *Computer Methods and Programs in Biomedicine*, *220*, p.106823.

[21]. Salah, Z., & Abu Elsoud, E. (2023). Enhancing Network Security: A Machine Learning-Based Approach for Detecting and Mitigating Krack and Kr00k Attacks in IEEE 802.11. *Future Internet*, *15*(8), 269.

[22]. https://www.kaggle.com/officialdataset/thyroid-hyper.

[23]. https://www.kaggle.com/officialdataset/thyroid-nodule.

[24]. https://www.kaggle.com/officialdataset/thyroid-ditis.

[25]. https://www.kaggle.com/officialdataset/thyroid-cancer.

[26]. Ryu, J. (2023). Adaptive Feature Fusion and Kernel-Based Regression Modeling to Improve Blind Image Quality Assessment. *Applied Sciences*, *13*(13), 7522.

[27]. Ayyarao, T.S.; Ramakrishna, N.S.S.; Elavarasan, R.M.; Polumahanthi, N.; Rambabu, M.; Saini, G.; Khan, B.; Alatas, B. War Strategy Optimization Algorithm: A New Effective Metaheuristic Algorithm for Global Optimization. *IEEE Access* **2022**, *10*, 25073–25105.

[28]. Das, L., Tripathy, A., & Nanda, S. (2023, September). Adaptive Wavelet Transform Based Protein Coding Region Prediction in DNA sequence. In *2023 1st International Conference on Circuits, Power and Intelligent Systems (CCPIS)* (pp. 1-6). IEEE.

[29]. GitHub—Microsoft/Swin-Transformer: This Is an Official Implementation for “Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows”. Available online: https://github.com/microsoft/Swin-Transformer (accessed on 29 December 2022).

[30]. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the 21st IEEE/CVF

International Conference on Computer Vision (ICCV), Virtual, 11–17 October 2021; pp. 10012–10022.

Vol. 20, No. 1. (2024)
E ISSN: 1672-2531