SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V12I3P116

Volume 12 Issue 3, 155-169, March 2025

Original Article

Reliable Detection of Polycystic Ovary Syndrome Using a Hybrid Deep Learning Approach: CNN-LSTM-GRU Integration

Sruthi SanilKumar

Applied Microbiology, SBST, Vellore Institute of Technology, Tamil Nadu, India.

Corresponding Author : sruthisanilkumar@outlook.com

Received: 20 December 2024 Revised: 11 March 2025 Accepted: 25 March 2025 Published: 31 March 2025

Abstract - *Polycystic Ovary Syndrome (PCOS) is a prevalent hormonal disorder in women of reproductive age categorized by the presence of numerous tiny cysts on the ovaries, greater levels of androgen and irregular menstrual cycles. PCOS detection involves identifying and categorizing ovarian health conditions employing medical imaging modalities. Accurate detection is essential for appropriate treatment and inhibition of related health problems. Challenges such as the intrinsic complexity of ovarian morphological characteristics and differences in image quality due to transformations in acquisition settings or noise significantly affect the accuracy of detection systems. Conventional methods severely depend on manual image examination and feature extraction, often leading to variations and limited reliability. This research focuses on creating a hybrid Deep Learning (DL) system for the unfailing detection of PCOS employing ultrasound images. The system was assessed on the Kaggle PCOS dataset containing 3,856 images categorized into "infected" and "not infected" cases. Preprocessing and data augmentation techniques were utilized to increase variability in data, followed by feature extraction operating the hybrid model. The proposed system merges Convolutional Neural Network (CNN), Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) architectures to collect spatial and temporal features expertly. The system achieved outstanding results with an accuracy of 98.50%, precision of 98.59%, recall of 98.55% and an F1 score of 98.48%. These results underscore the efficacy of the hybrid framework in responding to the challenges of PCOS detection, providing an efficient and better solution for clinical applications.*

Keywords - *Polycystic Ovary Syndrome, Deep Learning, Convolutional Neural Network, Ultrasound image, Long Short-Term Memory, Gated Recurrent Unit.*

1. Introduction

PCOS is a common endocrine condition that mostly influences women at their reproductive age, with far-reaching problems with hormonal equilibrium and normal health. An estimated range of 8-13% of women at their reproductive stage are affected by PCOS worldwide.

This disorder is characterized by various forms of hormonal imbalances and metabolic dysfunctions producing varied symptoms such as the occurrence of minute ovarian cysts, acne, irregular menstrual cycles, hirsutism and other androgen-related presentations depicted in Figure 1.

Hyperandrogenic hormone levels, which are normally present in lower amounts in women, interfere with the process of ovulation and lead to abnormal or absent menstrual cycles [1]. The syndrome is linked to other long-term health hazards like a 30-40% incidence of distributed glucose metabolism and an increased risk of depression or other psychological issues. PCOS is the primary etiology for anovulatory infertility and has also been associated with other serious conditions like thyroid dysfunction, obesity and endometrial or ovarian malignancies, highlighting the imperative need for early detection and intervention [2].

PCOS diagnosis usually relies on the Rotterdam criteria, which comprise three important markers: hyperandrogenism, irregular menstrual cycles and the existence of multiple ovarian cysts as detected by ultrasonography. Ultrasonography is the most accurate imaging technique.

However, it is highly dependent on the skill of radiologists and is also often hampered by image noise and observer variability. This constraint is especially evident in underdeveloped areas where skilled medical practitioners are inaccessible, resulting in delayed diagnosis and treatment [3]. Conventional Machine Learning (ML) and computational approaches to PCOS diagnosis have primarily relied on manual image classification and significant human intervention.

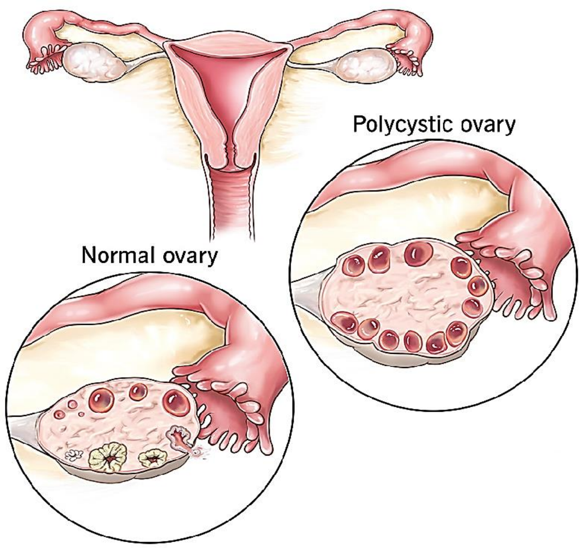


Fig. 1 Polycystic Ovary Syndrome (PCOS)

These methods have difficulty handling medical imaging complexities, specifically regarding noise and variability of the ultrasound data, and produce less consistent data. DL provides a revolutionizing functionality for processing medical images and delivers reliable means for the automated identification of sophisticated diseases such as PCOS [4]. A hybrid DL system in this study brings CNN, LSTM and GRU models together. This method leverages the advantage of every architecture to gather the spatial and temporal characteristics from ultrasound images, improving detection accuracy and robustness. By overcoming the limitations of conventional methods, the proposed model seeks to provide a reliable, efficient, and automated diagnostic tool for early-stage PCOS detection, ultimately enhancing patient outcomes and healthcare provision. The main contribution of the work includes

- To develop a hybrid DL framework that integrates CNN, LSTM and GRU architectures to efficiently extract spatial and temporal characteristics from ultrasound images for PCOS detection.
- The feasibility of the hybrid model presented will be examined utilizing evaluation metrics.
- To compare the designed hybrid model with existing methods to demonstrate its superiority in PCOS detection accuracy and reliability.

The remaining portion of the paper is arranged into four categories: Relevant research on ML and DL applications for PCOS detection is briefly reviewed in Section 2. The proposed methodology outlined in Section 3 presents the integration of CNN-LSTM-GRU architectures. Section 4 highlights the suggested hybrid framework's experimental outcomes and performance assessment. Section 5 concludes the research with key findings.

2. Literature Survey

2.1. PCOS Detection Using Machine Learning

Zad et al. (2024) [5] sought to create an ML-based prediction framework to aid early detection of PCOS among an outpatient category at risk using Electronic Health Records (EHR) data from 30,601 women aged 18-45 years at Boston Medical Center. The study utilized ML methods to build models achieving an AUC between 80% and 85%, with hormone levels and obesity as significant positive predictors.

A limitation noted was the applicability of the study to urban hospital-based populations and at-risk individuals, reducing its generalizability to broader or rural populations. Lim et al. (2023) [6] conducted a study to categorize and forecast PCOS using radial pulse wave parameters and ML models based on data from 459 individuals, comprising 316 in the PCOS cohort and 143 in the healthy cohort. The research applied seven supervised ML models, including KNN, Decision Trees (DT), LR, SVM, Voting, RF and LSTM, finding that the voting and LSTM models demonstrated 72.174% testing accuracy and 0.818 F1 score.

A limitation noted was the difficulty distinguishing noise from pulse signals during data collection. Batra and Nelson (2023) [7] created a Data-driven Computer-Aided Diagnostic System (DCADS) for the early detection of PCOS without clinical tests using the PCOS dataset from Kaggle. It applied the Synthetic Minority Oversampling Technique (SMOTE) for balancing and feature selection by correlation analysis and evaluated ML models, including RF, SVM and LR, achieving an accuracy of 92.024% with the RF model demonstrating promising results for non-clinical PCOS diagnosis. Suha and Islam (2023) [8] aimed to develop an Artificial Intelligence (AI) model for accurate PCOS detection using a modified ensemble ML technique that employed five traditional ML frameworks as base learners and a Gradient Boosting (GB) classifier as the meta-learner. The framework tested on patient symptom data with three types of feature selection strategies achieved 95.7% accuracy using PCA-selected top 25 features, surpassing other ML techniques.

The study noted a limitation on the small sample size. Bhardwaj and Tiwari (2022) [9] conducted a study to explore the application of ML algorithms for diagnosing PCOS using clinical data and a detailed preprocessing step to address data discrepancies. The study employed models such as Multi-Layer Perceptron (MLP), SVM with a Radial Basis Function (RBF) kernel, RF and XGBoost (XGB), achieving 93% accuracy with SVM and MLP supported by metrics like the ROC-AUC score and F1. Feature importance was assessed using a DT classifier with a Gini index identifying key factors like follicle size and prolactin levels. Rakshitha and Naveen (2022) [10] conducted a study to develop a hybrid and optimized ML approach for detecting PCOS using a combination of SVM with a linear kernel and LR (SVLR) optimized with the RMSprop optimizer. They collected 1600 datasets from a leading hospital in Bangalore Urban and demonstrated that their optimized-hybrid SVLR model achieved an accuracy of 89.03%, outperforming traditional algorithms like SVM, DT and RF. Dutta (2021) [11] aimed to develop an effective prediction model for early identification of PCOS, employing a dataset of 541 instances taken from the UCI repository with a significant class imbalance addressed through the SMOTE. The study employed LR, RF, DT, SVM and K-Nearest Neighbor (KNN), with SMOTE-based LR achieving an accuracy of 97.11% while SMOTE-based RF showed the lowest execution time.

Despite its promising results, the study acknowledged challenges related to imbalanced data that impact the model's performance. Bharati et al. (2021) [12] concentrated on developing a data-driven method for diagnosing PCOS in women using a dataset from the Kaggle repository, which included 177 samples with 43 features. The study employed univariate feature selection and feature elimination methods to determine significant predictors ranking the Follicle-Stimulating Hormone (FSH) to Luteinizing Hormone (LH) ratio as the most critical and applied ensemble ML models including soft and hard voting and CatBoost, with 5, 10 and 20-fold cross-validation. The results demonstrated that soft voting achieved an accuracy of 91.12%, confirming the efficacy of ensemble learning for PCOS detection.

2.2. PCOS Detection Using Deep Learning

Kumar and Varadarajan (2024) [13] conducted a study aimed at creating prediction systems for early identification of PCOS to address related health risks using both image and text datasets. The study employed ensemble learning techniques, including LR, RF and SVM, achieving an 89% accuracy with an AUC of 0.83 and advanced DL models integrating CNN and LSTM networks for an enhanced fully connected neural network that reached a 96.07% accuracy with minimal loss. Umapathy et al. (2024) [14] conducted a study to detect PCOS by implementing YOLOv8 for ovarian follicle detection and segmenting them utilizing a hybrid fuzzy c-means-based active contour approach. It used two datasets: one dataset with 50 normal and 50 PCOS subjects and another dataset with 100 PCOS and 100 normal subjects.

The study extracted features through Gray-level Co-occurrence Matrices (GLCM). It tested several ML and DL classifiers, including a custom-built Follicles Net (F-Net) model that achieved a superior classification accuracy of 95% and 97.5% for the respective datasets. The study's limitation was its lack of clinical validation in real-world healthcare environments. Almoudi et al. (2023) [15] proposed improving the diagnosis of PCOS based on a DL strategy employing a dataset with ovary ultrasound images and clinical patient information. Inception and MobileNet models were used, reaching 84.18% accuracy using the Inception model for image diagnosis and 82.46% with a fusion model integrating image and clinical data. One of the study's main limitations was the restricted computational power that could be used to carry out the empirical analysis. Fan et al. (2023) [16] created a DL system, Ocys-Net, for the diagnosis and classification of ovarian cysts to help with quick diagnosis and alleviate doctors' workloads.

The research utilized an ovarian cyst dataset, attaining a classification accuracy of 95.93% through the use of a reverse bottleneck structure scheme and an Efficient Channel Attention (ECA) module for improved feature extraction. The research underscored the constraints in medical image analysis as a result of the lack of theoretical knowledge that inhibits efficient visual feature extraction and processing. Wenqi Lv et al. (2022) [17] undertook research to develop an automated DL technique for PCOS detection grounded on scleral alteration analysis with the help of 721 full-eye images from Chinese women consisting of 388 PCOS-affected roles.

The approach entailed segmenting scleral images using an enhanced U-Net to extract deep features using a ResNet method and applying a multi-instance classification method with an AUC of 0.979 and an accuracy of 92.9%. The research was limited by a relatively small dataset and difficulties interpreting certain visualization results. Hosain et al. (2022) [18] sought to develop and contrast DL models for identifying PCOS from ovarian ultrasound images to aid in early diagnosis. PCONet was designed by the study as a bespoke CNN model, and a pretrained Inception V3 was fine-tuned by the study employing Transfer Learning (TL).

The models were evaluated on a separate test set to prevent bias. PCONet outperformed Inception V3 with an accuracy of 98.12% compared to 96.56%, demonstrating its potential for clinical application. Srivastava (2020) [19] conducted a study aimed at detecting ovarian cysts using ultrasound images, recognizing their impact on female reproductive health and potential risks such as torsion and cancer. The study utilized a fine-tuned VGG-16 DL method with modifications to the last four layers trained on a custom dataset of ultrasound images collected from various women.

This approach achieved an accuracy of 92.11%. The existing works are summarized in Table 1. Although numerous studies have applied ML and DL techniques to detect PCOS, there are still several issues in the existing research. Many studies have focused on specific data types, such as electronic health records clinical parameters, but few have integrated diverse data sources to provide a holistic diagnostic approach.

While ML algorithms like RF, SVM and GB have shown moderate to high accuracy, their dependence on feature selection and handcrafted features limits their ability to generalize to unseen datasets. Although DL models effectively handle complex patterns in medical images, they often require large datasets for training, which is challenging given the sensitive and limited nature of medical data.

Table 1. Summary of existing studies

Author & Year	Model Used	Dataset	Accuracy & Key Findings	Limitations
Zad et al. (2024) [5]	LR, SVM, RF, GBT	EHR data (30,601 women)	80-85% (AUC): Identified hormone levels and obesity as key predictors	Limited to urban, hospital-based populations
Lim et al. (2023) [6]	KNN, DT, LR, SVM, Voting, RF, LSTM	Radial pulse wave data (459 subjects)	72.17%: Voting and LSTM models performed best	Difficulty distinguishing noise from pulse signals
Batra & Nelson (2023) [7]	RF, SVM, LR (with SMOTE)	Kaggle PCOS dataset	92.02%: Feature selection improved accuracy	Dependent on feature engineering
Suha & Islam (2023) [8]	Modified ensemble ML (GB as meta-learner)	Patient symptom data	95.7%: PCA-based feature selection improved results	Small dataset size
Bhardwaj & Tiwari (2022) [9]	SVM (RBF), RF, XGBoost, MLP	Clinical data	93%: Identified follicle size and prolactin levels as key features	Feature selection impacts generalizability
Rakshitha & Naveen (2022) [10]	SVM (Linear) + LR (SVLR)	Clinical dataset (1600 samples)	89.03%: Optimized hybrid SVLR model improved detection	Traditional ML models are still used
Dutta (2021) [11]	LR, RF, DT, SVM, KNN (with SMOTE)	UCI dataset (541 instances)	97.11%: SMOTE improved LR model performance	Class imbalance issues
Bharati et al. (2021) [12]	Soft & Hard Voting, CatBoost	Kaggle dataset (177 samples)	91.12%: FSH/LH ratio was the most critical predictor	Small dataset, overfitting risk
Kumar & Varadarajan (2024) [13]	CNN-LSTM	Image and text datasets	96.07%: Improved feature extraction	Needs further validation
Umapathy et al. (2024) [14]	YOLOv8 + Hybrid Fuzzy C-Means	Two datasets (100 PCOS, 100 normal)	95%-97.5%: Used follicle segmentation for better classification	Lacks clinical validation
Almoudi et al. (2023) [15]	Inception, MobileNet	Ovary ultrasound images	84.18%: Fusion of image and clinical data improved detection	Limited computational resources
Fan et al. (2023) [16]	Ocys-Net (Reverse Bottleneck + ECA)	Ovarian cyst dataset	95.93%: Enhanced feature extraction with attention mechanisms	Lack of theoretical knowledge for image analysis
Wenqi Lv et al. (2022) [17]	Improved U-Net + ResNet	Scleral images (721 samples)	92.9% (AUC 0.979): Multi-instance classification improved accuracy	Small dataset, interpretation challenges
Hosain et al. (2022) [18]	PCONet (Custom CNN) + Inception V3 (TL)	Ovarian ultrasound images	98.12% (PCONet), 96.56% (Inception): PCONet outperformed transfer learning	Model bias risk
Srivastava (2020) [19]	Fine-tuned VGG-16	Custom ultrasound dataset	92.11%: Effective for ovarian cyst detection	Dataset variability limits generalization

Studies employing hybrid models such as CNN-LSTM or ensemble techniques have demonstrated promise, yet they lack real-world clinical validation and often overlook the importance of incorporating temporal dependencies in PCOS-related symptoms and imaging dataAnother limitation is the lack of explainability in DL models, which poses a challenge in gaining the trust of clinicians and patients. Many studies use imbalanced datasets or small sample sizes, leading to potential overfitting and reducing the reliability of the proposed models. Considering the efficiency of DL frameworks in PCOS detection, real-time implementation in clinical settings remains a challenge. Major challenges involve the high computational requirements for explainable AI for clinical use and the variability of ultrasound imaging conditions. Addressing these challenges requires developing stable hybrid frameworks that leverage varied data types and ensure interpretability. They are proven in real-world clinical settings to provide broad applicability and consistency.

3. Materials and Methods

Predictive modeling for the detection of PCOS employed a hybrid DL model with CNN, GRU and LSTM layers from ultrasound images. The methodology workflow is depicted in Figure 2. The input dataset is an ultrasound image that goes through preprocessing and augmentation to enhance the superiority and diversity of the data. Following preprocessing, a testing set and a training set were prepared from the data. A hybrid model is trained by means of the training set. CNN layers are employed to extract features, and GRU layers are used to determine sequential dependencies from the data. LSTM layers are then used to collect long-term dependencies, improving the system's ability to interpret complex patterns. The system performance is then measured after training using the testing set to analyze its accuracy and functionality. The trained model generates a final predicted output, classifying each input as "infected" or "not infected," which offers insights into PCOS detection from ultrasound images.

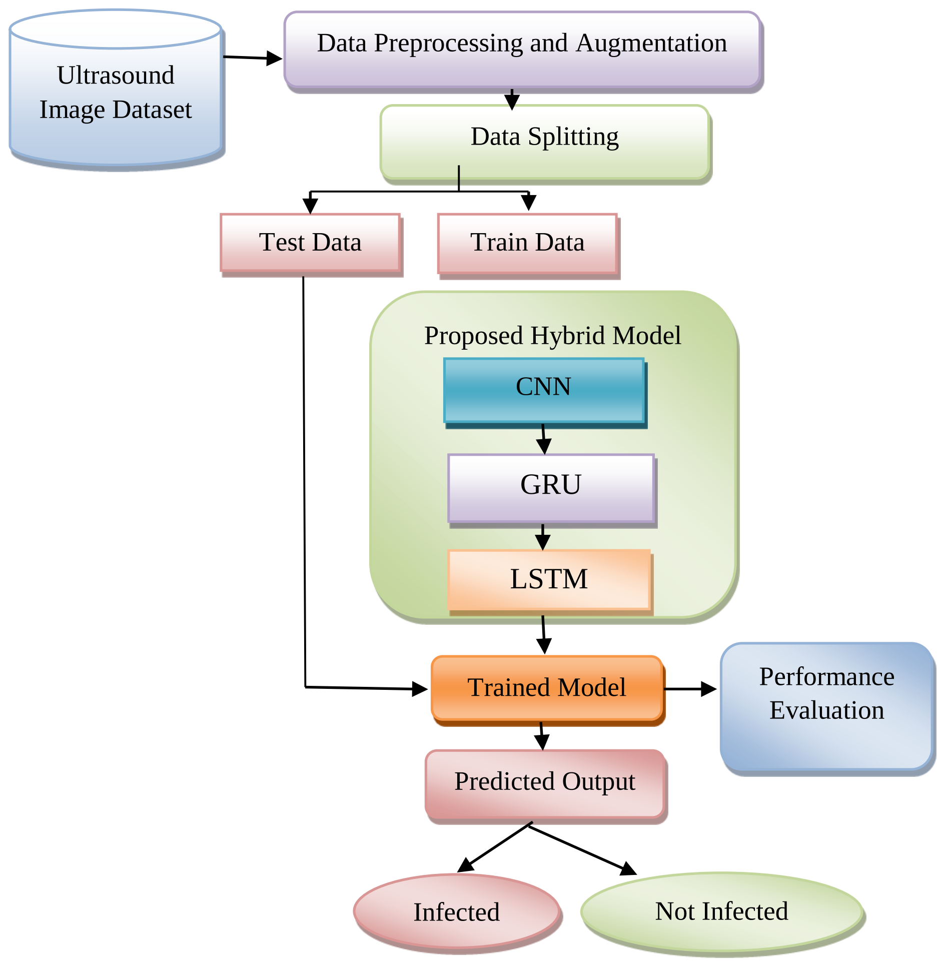


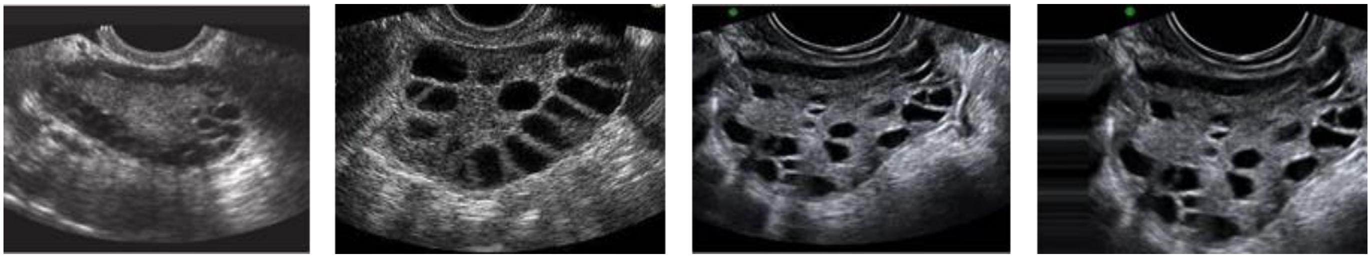
Fig. 2 Block diagram of suggested method

3.1. Dataset Description

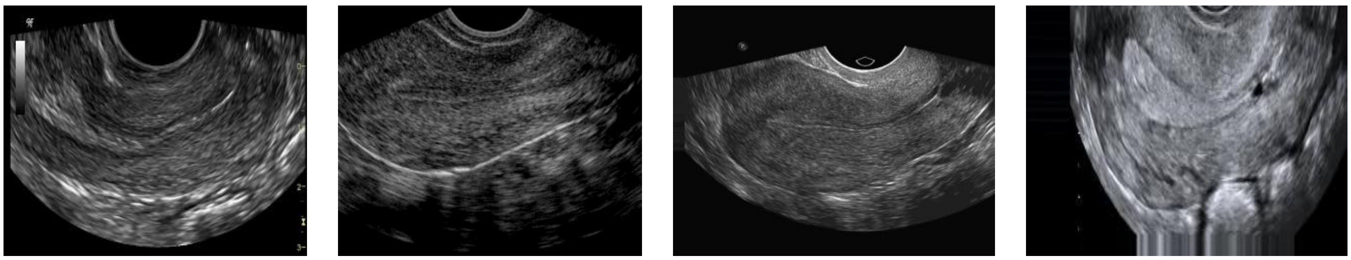
The data for predicting PCOS from ultrasound images is attained from Kaggle's database [20]. The dataset encompasses a total of 3,856 images. It is divided into two primary subfolders, 'train' and 'test', which were further split into two categories ", infected" and "not infected", as depicted in Figure 3. The 'infected' category consists of ultrasound images of PCOS-diagnosed ovaries, and the 'not infected' category consists of healthy ovaries images free from the disease.This dataset facilitates improved PCOS detection through automated means and aids in the early diagnosis by medical imaging analysis.

3.2. Data Preprocessing and Augmentation

Preprocessing is also important in improving the DL model's efficiency for PCOS detection from ultrasound images. It normalizes the input through techniques like resizing the images to the same dimension of 224*224 pixels and pixel value normalization to values between 0 and 1. The model can process the images better and learn from uniformly scaled data. Data augmentation advances the capability of the model to generalize by enhancing the variability of the training dataset. The augmentation methods used include random shear transformations (±20 degrees), zooming (up to 20%), vertical and horizontal flipping and rotation (up to 30 degrees).



(a) Infected images



(b) Not infected images

Fig. 3 Sample images from the dataset

These techniques help the model learn variations in ovarian structures and improve robustness against overfitting. A stratified split was used to maintain class balance in the training and validation sets, with 85% of the dataset applied for training and 15% reserved for testing, which enables the system to effectively classify the images

3.3. Model Development

The three models employed in this study are CNN, GRU and LSTM. CNNs are specialized for image data, utilizing convolutional layers to extract spatial features, pooling layers to decrease the dimensionality and Fully Connected (FC) layers for classification. GRU is a variant of Recurrent Neural Networks (RNN) that excels at handling sequential data by using reset and update gates to balance the retention of past and current information. LSTM is another RNN variant that uses memory cells to handle long-term dependencies in sequential data and gating mechanisms (input, forget and output gates), ensuring effective information flow across time steps. Altogether, these models provide a robust framework for processing and analyzing both spatial and temporal data in PCOS detection tasks.

3.3.1. Convolutional Neural Network

CNN are highly effective DL models created for analyzing structured grid-based data, particularly images and sequential data. As shown in Figure 4 it consists of several key elements which combine together to capture and process features from raw data. Convolutional layers modify the input by applying filters or kernels to identify local patterns such as edges, textures, and other distinct features. By sliding these filters along the data, CNN learns both low-level and abstract features as the data progresses through successive layers. The pooling layer decreases the resolution of feature maps by downsampling and compacting the data to retain essential information, lowering computational complexity and minimizing overfitting by focusing on key patterns. FC layers analyze these learned features to make high-level predictions such as classifications. Its architecture allows flexibility in configuring layer counts, filter sizes and activation functions, making them highly adaptable to diverse tasks [21]. Convolution generates a feature map by sliding a filter over an input image, conducting element-wise multiplication at each position, and adding the results. This process is mathematically expressed in Equation (1).

$$(H * K)(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} K(x+m, y+n)\, H(m, n) \quad (1)$$

Where H represents the input image, K presents the convolutional kernel, x and y are the points in the output feature map and m and n are the indices iterating over the dimensions of the kernel K. After the convolution operation, the resulting feature map is subjected to a non-linear modification using an activation function, commonly the Rectified Linear Unit (ReLU). This transformation is mathematically described in Equation (2).

$$f(x) = \max(0, x) \quad (2)$$

The pooling layers are used to decrease the dataimensionality while keeping its important features. Max pooling is widely regarded as one of the most commonly used techniques where a specified area of the feature map with maximum value is selected. This operation is mathematically represented by Equation (3).

$$Z(x, y) = \max_{(m,n)\epsilon S} F(m, n) \quad (3)$$

The region S, represents the area over which maxpooling is applied.

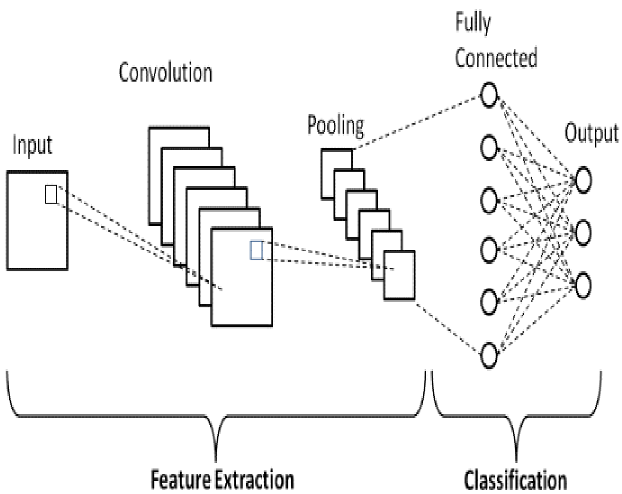


Fig. 4 Basic architecture of CNN

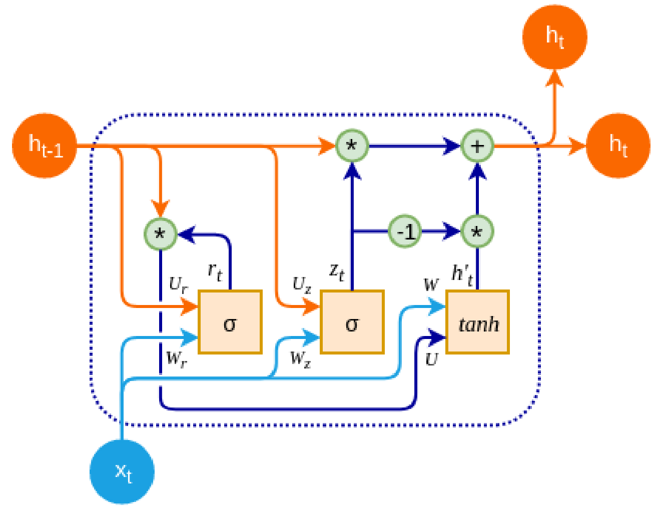


Fig. 5 Basic architecture of GRU

After pooling, the feature maps are flattened into a 1D vector, which is then given into FC (dense) layers, which are assigned for performing classification or regression tasks. The computation performed by the dense layer is represented by Equation (4).

$$w = B \cdot x + b \tag{4}$$

Where B demonstrates the weight matrix, the input vector from the previous layer is denoted as x , b is the bias term, and w is the FC layer output. The output layer then generates a probability distribution using the softmax function as given in Equation (5).

$$\sigma(w)_x = \frac{e^{w_x}}{\sum_{y=1}^{K} e^{w_y}} \tag{5}$$

Where, $\sigma(w)_x$ is the probability of class x, and K denotes all of the classes.

3.3.2. Gated Recurrent Unit

GRU is developed for sequential data processing, producing an output at each time step to capture complex temporal dependencies. Its architecture, shown in Figure 5, consists of two key gates, namely the reset gate, which identifies the proportion of the previous hidden state that should be "forgotten", and the update gate, which regulates the combination of new input and past information to form the new hidden state. It computes candidate activation by combining current input with the previous state and compresses this using a $tanh$ function for a range between -1 and 1. The final output layer uses the updated hidden state for predictions, representing sequential values or class probabilities [22]. The model processes the current input x_t at each time step t along with the hidden state from the preceding time step h_{t-1}.This generates a new hidden state h_t which is subsequently transformed into the next step for ongoing computation.

The update gate is denoted as z_t which combines the roles of the traditional input and forget gates controlling how much of the past state is retained versus how much new information is added. This mechanism effectively balances memory retention and the integration of new data, as illustrated in Equation (6).

$$z_t = \sigma(W_z * [h_{t-1}, x_t]) \tag{6}$$

WhereW_z, corresponds to the update gate z_t's trainable weight matrix. The reset gate plays a crucial part in identifying the degree to which the preceding state's output is integrated into the current state. This influences the computation in Equation (7), which effectively balances the integration of past and current information to update the hidden state.

$$r_t = \sigma(W_r * [h_{t-1}, x_t]) \tag{7}$$

WhereW_r, corresponds to the trainable weight matrix of the reset gate r_t. The candidate activation vector $\tilde{h}_t$, as shown in Equation (8), was derived by combining the current input x_t with an altered form of the preceding hidden state, which is controlled by the reset gate. This operation allows the network to selectively incorporate useful past information while calculating the candidate of the new hidden state.

$$\tilde{h}_t = \tanh(W_h * [r_t * h_{t-1}, x_t]) \tag{8}$$

HereW_h, represents an additional weight matrix involved in this computation that enables the smooth combination of past and current data to update the hidden state efficiently. The new hidden state h_t as described in Equation (9), was computed by combining the candidate activation vector with the previous hidden state, where the contribution is weighted by the update gate.

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{9}$$

3.3.3. Long Short-Term Memory

LSTM is designed to overcome the limitation of RNN in capturing sequential data's long-term dependencies. LSTM networks provide a new architecture, as presented in Figure 6, consisting of memory cells and gating mechanisms, namely the output, input, and forget gates that regulate the flow of information. The architecture enables LSTMs to store important data for long durations and effectively deal with vanishing and exploding gradient issues prevalent in traditional RNNs. By selectively remembering or forgetting information, LSTM networks excel in tasks requiring context over sequences, such as language modeling, time-series prediction and speech recognition [23]. The basic formula to calculate the hidden state h_t at time step t is described by Equation (10), which calculates the hidden state according to present input and previous state information.

$$h_t = f(Mx_t + Wh_{t-1}) \tag{10}$$

Where f denotes a non-linear activation function, including $tanh()$ or ReLU. This function handles x_t which is the current input state and h_{t-1} is the preceding hidden state at time steps t and $t-1$. The LSTM's three gates control the procedure of adding, deleting and utilizing memory content accordingly. The forget gate (f_t) in an LSTM handles the cell state by determining which data should be retained or deleted. It compares the earlier hidden state to the present input using a sigmoid activation function, and those outputs vary between 0 and 1. If the output is close to 0, it is denoted that the information can be forgotten. If an output is near 1, it determines that the information should be preserved. This selective filtering directly modifies the cell state through element-wise multiplication, effectively removing irrelevant data and maintaining only essential information, as shown in Equation (11).

$$f_t = \sigma(M_f x_t + W_f h_{t-1}) \tag{11}$$

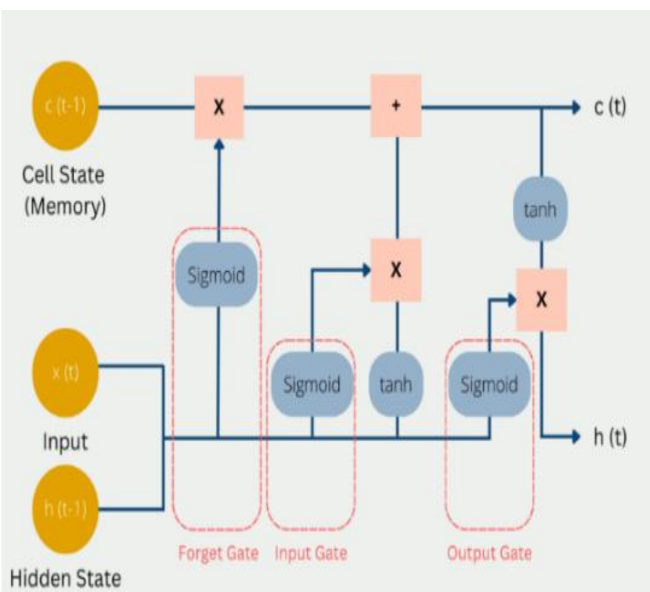


Fig. 6 Basic architecture of LSTM

Where M_f and W_f are the weight matrix for the input state and hidden state. The input gate (i_t) evaluates the relevance of the current input by combining it with the previous hidden state and a corresponding weight matrix. The important information is then incorporated into the cell state, modifying to reflect the new long-term memory, which is used in subsequent iterations of the network as depicted in Equation (12).

$$i_t = \sigma(M_i x_t + W_i h_{t-1}) \tag{12}$$

Where M_i and W_i denotes the weight linked to the present input and previous hidden state. The output of an LSTM network is determined by the output gate (o_t). The cell state, holding the long-term information, was modified by a $tanh$ function, which refined the selection process. This mechanism allows the network to maintain relevant data over time and produce outputs of the given input sequence represented in Equation (13).

$$o_t = \sigma(M_o x_t + W_o h_{t-1}) \tag{13}$$

Where M_o and W_o represents the weight linked to the current input and preceding hidden state. LSTM utilizes the cell state c_t that functions as the model's long-term memory and the hidden state h_t representing short-term memory for storing information. The cell state keeps the important data across extended sequences, and the hidden state extracts more information from recent computational steps. This dual memory allows LSTM to efficiently coordinate and utilize both current inputs and important past information, processing the present input. x_t, the previous cell state c_{t-1} and the preceding hidden state h_{t-1}. Equation (14) calculates the cell state as follows:

$$\hat{c}_t = \tanh(M_c x_t + W_c h_{t-1}) \tag{14}$$

The updated internal memory is calculated in Equation (15).

$$c_t = \hat{c}_t * i_t + c_{t-1} * f_t \tag{15}$$

As a result, the output gate and memory cell state are used to determine the hidden state at time step t in Equation (16) as follows.

$$h_t = o_t * \tanh(c_t) \tag{16}$$

3.4. Proposed Hybrid Model

The proposed hybrid DL network combines CNN with GRU and LSTM layers to extract spatial and temporal dependencies in the data. The model begins with the CNN branch, which performs feature extraction from the input images, employing convolutional layers with ReLU activation to identify spatial patterns. A max-pooling layer down a sample of the feature maps, reducing their dimensionality and dropout, is applied to prevent overfitting during training. This output is then transferred through an FC layer with ReLU activation before being further processed by a dense layer with

32 units. The second branch of the model incorporates the GRU layers to extract sequential dependencies. The input of the framework is given to convolutional layers to collect characteristics tailed by a max pooling, dropout and reshaping to prepare it for sequential processing. This allows the model to capture patterns over time, which is particularly useful for detecting long-term dependencies in ultrasound images. The third branch incorporates an LSTM layer to retain long-term dependencies. Similar to GRU, the features are extracted through a convolutional layer followed by max pooling and dropout.

After reshaping, the data is run through an LSTM layer that processes the temporal data, allowing the model to capture long-term temporal patterns relevant to PCOS detection. Then, the CNN, GRU and LSTM outputs are concatenated and passed through a dropout layer to reduce overfitting. The concatenated output is then processed by a dense layer with ReLU activation followed by a final output layer with a softmax activation function producing a probability distribution across the "infected" and "not infected" classes. This hybrid architecture provides a powerful system for accurately identifying PCOS from ultrasound images. Figure 7 presents the architecture of the suggested study. The algorithm for the suggested system is shown below.

Algorithm: PCOS Detection Using Hybrid CNN-GRU-LSTM Model

Input: Ultrasound images of ovaries
Output: PCOS detection model (infected or not infected)

Begin:
Load and preprocess data:
1. Collect dataset: $C = \{(P_i, Q_i)\}_{i=0}^{N-1}$, were P_i is an ultrasound image and $Q_i \in \{0, 1\}$ (1: infected, 0: not infected).
2. Preprocess:
    - Resize: $P_i \rightarrow P_i' \in R^{224\times224}$
    - Normalize: $P_i' \rightarrow \frac{P_i' - \mu}{\sigma}$
    - Data Augmentation: $P_i' \rightarrow \{P_i''\}$(Shear, Zoom, Flip (horizontal and vertical), Rotation)

Define CNN-GRU-LSTM Model:
1. Input: $224 \times 224 \times 3$
   CNN Branch
   Conv2D (16, (3,3), activation='relu')
   MaxPooling2D (pool size= (2, 2))
   Dropout (0.9)
   Flatten ()
   Dense (8, activation='relu')
   Dense (32)
   GRU Branch
   Conv2D (8, (3,3), activation='relu')
   MaxPooling2D (pool size= (2, 2))
   Dropout (0.9)
   Flatten ()
   Reshape for GRU
   GRU (4)
   Flatten ()
   LSTM Branch
   Conv2D (8, (3,3), activation='relu')
   MaxPooling2D (pool size= (2, 2))
   Dropout (0.9)
   Flatten ()
   Reshape for LSTM
   LSTM (4)
   Flatten ()
   Concatenate ()
   Fully Connected Layers:
   Dropout (0.9)
   Dense (8, activation='relu')
   Dense (2, activation='softmax')
2. Compile the model X:
   optimizer=Adam ()
   learning rate=0.01
   loss function=binary _crossentropy
   metrics=[accuracy]
3. Train the model X:
   Fit the model: X. fit (P_{train} , Q_{train} ,validation_data= (P_{val}, Q_{val}), batch size=32, epochs=50).
4. Evaluate the model X:
   Evaluate: X. eval ((P_{test} , Q_{test}), where metrics include accuracy, recall, precision and F1 Score.

Save the Model
End

3.5. Software and Hardware Setup

The proposed system was built using the Google Colaboratory platform, utilizing Python and the Keras model for implementation. The Colab notebooks use 64-bit Windows 10 and come with a Graphics Processing Unit (GPU), 68.50 GB of storage and 12.75 GB of RAM pre-installed with TensorFlow. Python's flexibility, straightforward syntax and extensive library support make it an ideal choice for this study. The hyperparameters, pre-set configurations that influence a DL model's learning process, were determined through empirical methods as summarized in Table 2.

The framework that produced the optimal performance in classification was determined by comparing and analyzing several sets of variables. A 0.9 dropout rate was chosen to reduce overfitting due to the framework's complex architecture. Experiments with lower dropout rates resulted in higher variance and overfitting during validation.

The selected dropout rate ensures better generalization on unseen data. The learning rate was optimized to balance convergence speed and stability, while the number of epochs was determined based on performance improvements observed during training.

Table 2. Hyperparameters of the hybrid model

Hyperparameters	Values
Activation Function	ReLU, Softmax
Learning rate	0.01
Dropout	0.9
Number of Epochs	50
Optimizer	Adam
Loss Function	Binary Crossentropy
Batch Size	32

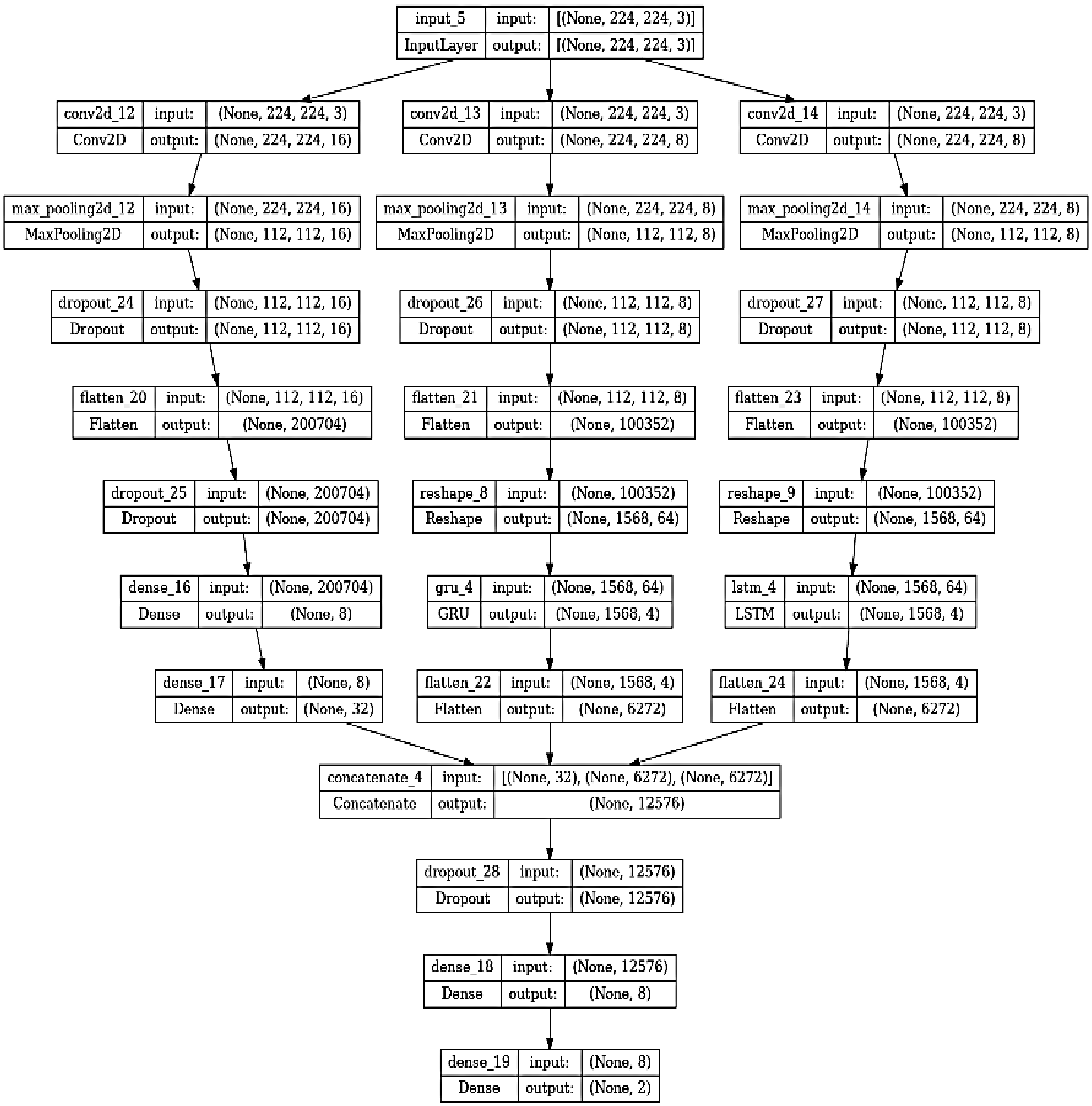


Fig. 7 Proposed model architecture

The optimizer was selected for its efficiency in handling complex gradient updates, and the batch size was adjusted to maintain a trade-off between computational efficiency and model accuracy. Activation functions were chosen to introduce non-linearity and improve classification performance, while the loss function was selected to effectively minimize classification errors in PCOS detection.

4. Results and Discussion

The accuracy plot presents the system's efficiency by tracking the proportion of correct predictions over time, and the loss plot monitors the error among the actual and predicted values during training. Both visualizations are essential for evaluating the model's learning progression with higher accuracy and lower loss, indicating better performance. In this study, these plots provide important insights into the model's capability to detect PCOS based on the dataset's features. Figure 8 displays the accuracy and loss plots for the suggested framework. At the initial epoch, the system accuracy is relatively low, starting around 86%. It determines that the model is just beginning to learn and understand the data patterns. The accuracy gradually improves during the initial training phase, showing that the framework successfully learns basic features from the training set. By the final epoch at around 50, the training accuracy becomes significantly higher, reaching approximately 98.5%. This suggests that the model has effectively learned complex data relationships and patterns. When considering the loss of the system, it is relatively high at the initial epoch, as shown by the training loss starting above 0.5. The loss decreases rapidly over the early epochs as the model adjusts its biases and weights learning to represent the underlying structure of the data. By the final epoch, the system loss is significantly minimized and reaches around 0.1. This notable reduction in loss throughout the training process determines that the framework is efficiently improving its accuracy and reducing its error rate. Evaluation metrics are crucial for evaluating DL framework performance, offering information about their predictive and classification capabilities. Metrics commonly used include accuracy, precision, recall, and the F1 score, each providing an alternative viewpoint on the model's performance. These metrics are crucial for identifying issues like overfitting, underfitting or class imbalance, and they guide improvements during the training phase. Equations (17)-(20) define the formulas.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{17}$$

$$Recall = \frac{TP}{TP+FN} \tag{18}$$

$$Precision = \frac{TP}{TP+FP} \tag{19}$$

$$F1\ Score = 2 * \frac{(Recall*Precision)}{(Recall+Precision)} \tag{20}$$

Where TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative. Figure 9 represents the graphical illustration of the suggested hybrid model that demonstrates excellent performance with 98.50% accuracy. Precision and recall are 98.59% and 98.55%, respectively, which presents the strong ability of the framework to make precise positive predictions while effectively identifying TP. The F1 score of 98.48% demonstrates a strong overall classification performance, correctly predicting positives and reducing FP and FN. Overall, the hybrid model exhibits a strong ability to classify data with high reliability and minimal errors.

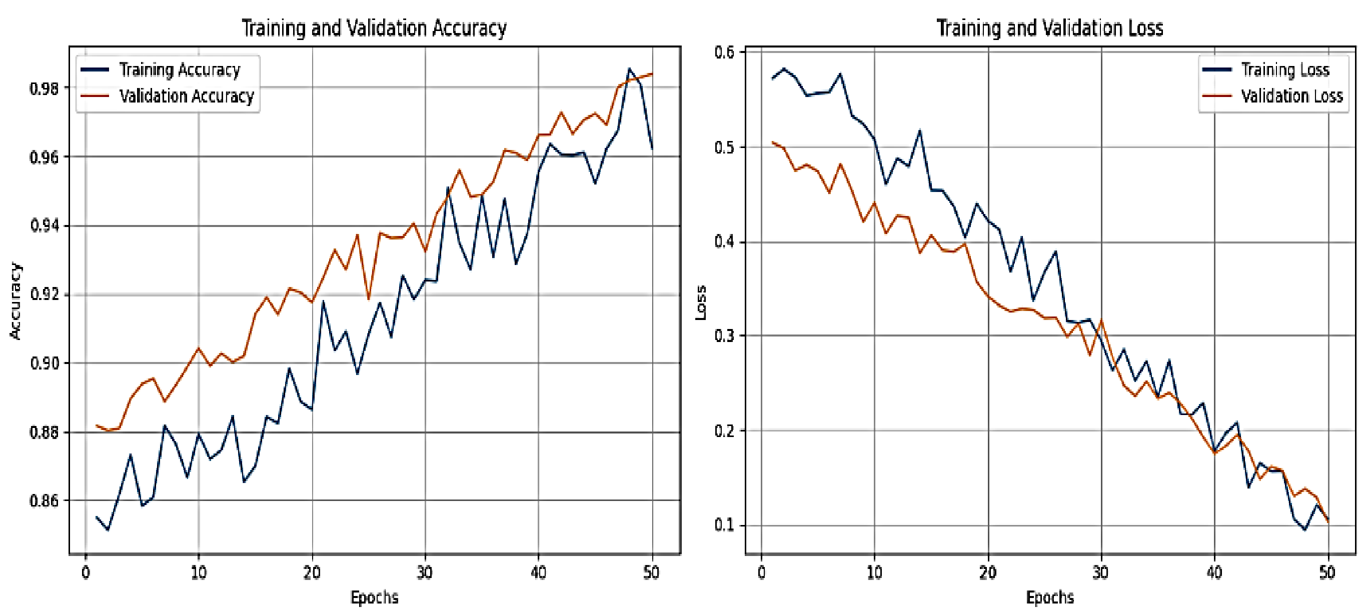


Fig. 8 Accuracy and loss plot of the hybrid method

A tool utilized to examine the efficiency of a classification framework is a confusion matrix that displays the number of TP, TN, FP and FN. Figure 10 illustrates the confusion matrix by comparing the actual outcomes with the predicted ones. The matrix shows that the model correctly predicted 169 out of 171 samples as "not infected" and 170 out of 171 samples as "infected". There were 2 healthy ovaries misclassified as "infected" and 1 infected ovary misclassified as healthy. Overall, the model demonstrates high accuracy with a strong balance between sensitivity and specificity. An image selected randomly from the dataset was evaluated utilizing the proposed hybrid framework that accurately classified it as "infected." As demonstrated in Figure 11, this correct classification emphasizes the effectiveness of the system and its capacity to reliably identify and categorize images within the dataset.

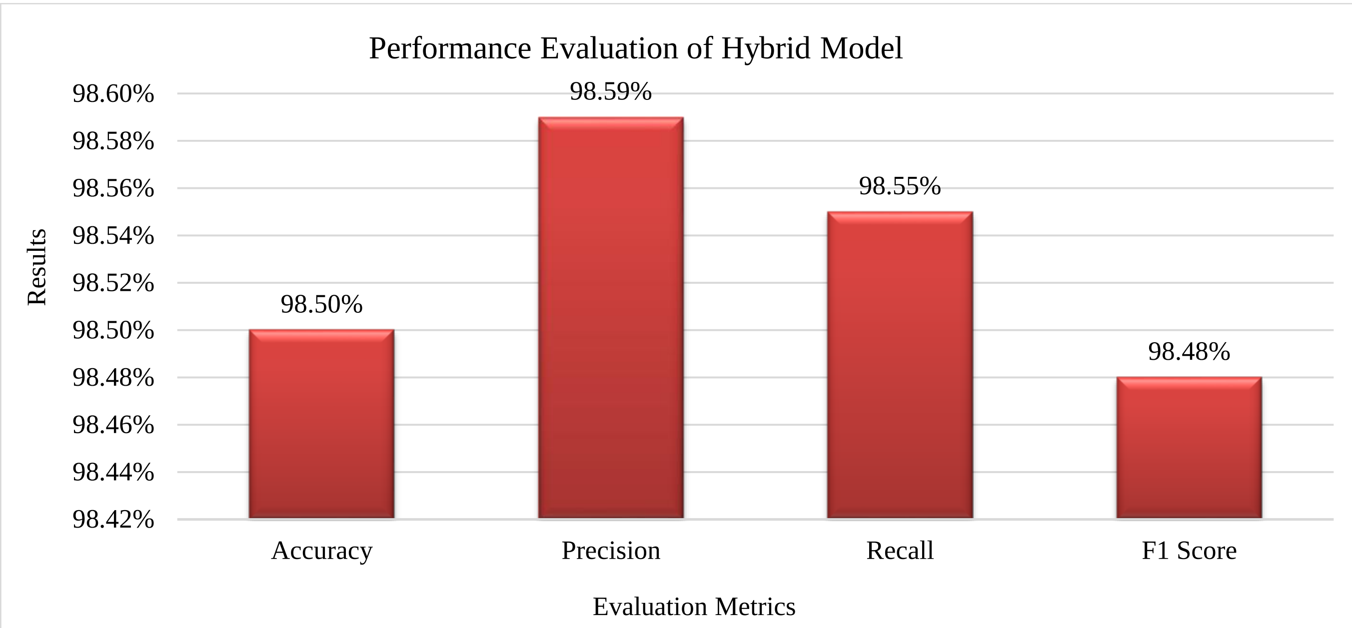


Fig. 9 Performance evaluation of the proposed hybrid model

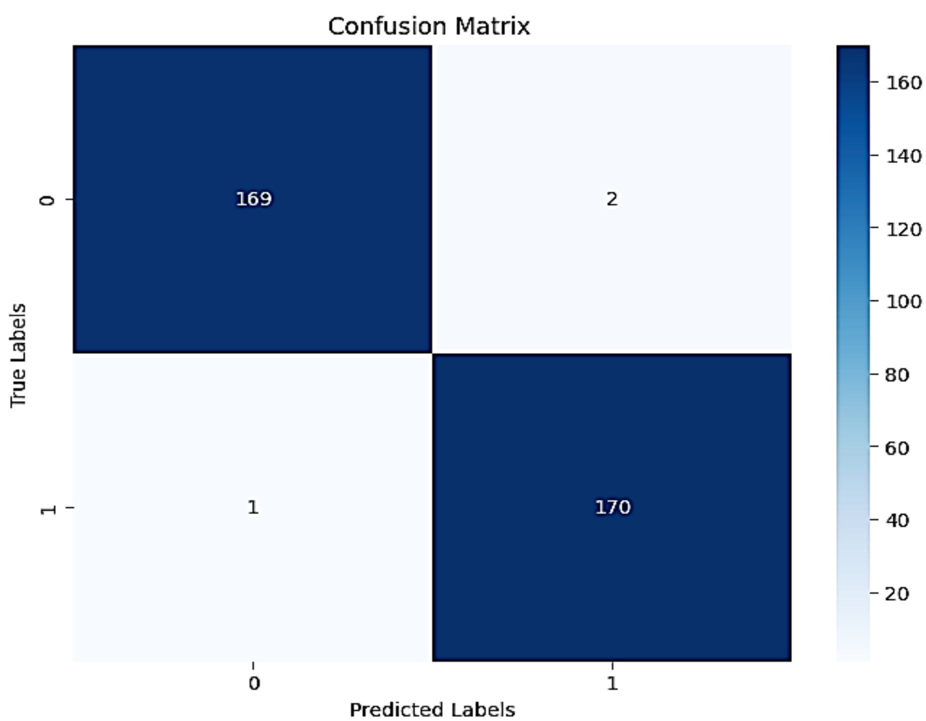


Fig. 10 Confusion matrix of the proposed hybrid model

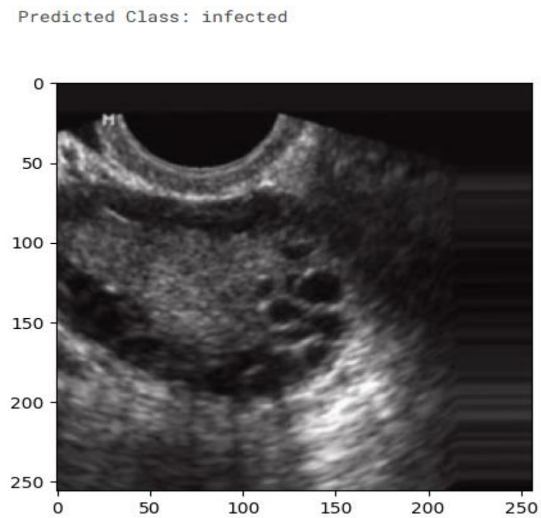


Fig. 11 Predicted output of PCOS

Table 3 provides a comparative analysis of the accuracy attained by various models in detecting PCOS across different datasets. The proposed hybrid CNN-GRU-LSTM model exhibits an exceptional accuracy of 98.50%, outperforming all existing frameworks. The voting-LSTM model obtained an accuracy of 72.17% on radial pulse wave data, and Inception attained 84.18% using ovary ultrasound images. Other methods like SLVR and RF attained an accuracy of 89.03% and 92.02%. It showed significant improvements but fell short compared to the proposed model. Even advanced models like U-Net, with an accuracy of 92.9%, and Ocys-Net, obtained an accuracy of 95.93%. The proposed hybrid model's superior performance can be attributed to the integration of CNN for feature extraction combined with the sequential modeling strengths of GRU and LSTM. This hybrid architecture effectively captures both temporal and spatial dependencies in ultrasound imaging data, providing a more thorough analysis than single models or less complex architectures. Using ultrasound image data enhances the robustness of the model, utilizing high-quality diagnostic imaging for accurate detection. Higher accuracy indicates that the proposed model could provide reliable and precise PCOS detection, ensuring a more effective solution than existing methods. Figure 12 graphically illustrates the accuracy of comparing the hybrid model with the existing approaches.

Table 3. Accuracy comparison of proposed and existing methods

Authors & References	Models	Datasets	Accuracy
Lim et al. [6]	Voting-LSTM	Radial pulse wave data	72.17%
Almoudi et al. [15]	Inception	Ovary ultrasound images	84.18%
Rakshitha & Naveen [10]	SLVR	Clinical dataset	89.03%
Batra & Nelson [7]	RF	Kaggle PCOS dataset	92.02%
Wenqi Lv et al. [17]	U-Net	PCOS dataset	92.9%
Bhardwaj & Tiwari [9]	SVM	Clinical data	93%
Fan et al. [16]	Ocys-Net	Ovarian cyst dataset	95.93%
Kumar & Varadarajan [13]	CNN-LSTM	Image and text datasets	96.07%
Proposed Hybrid CNN-GRU-LSTM Model		**Ultrasound image dataset**	**98.50%**

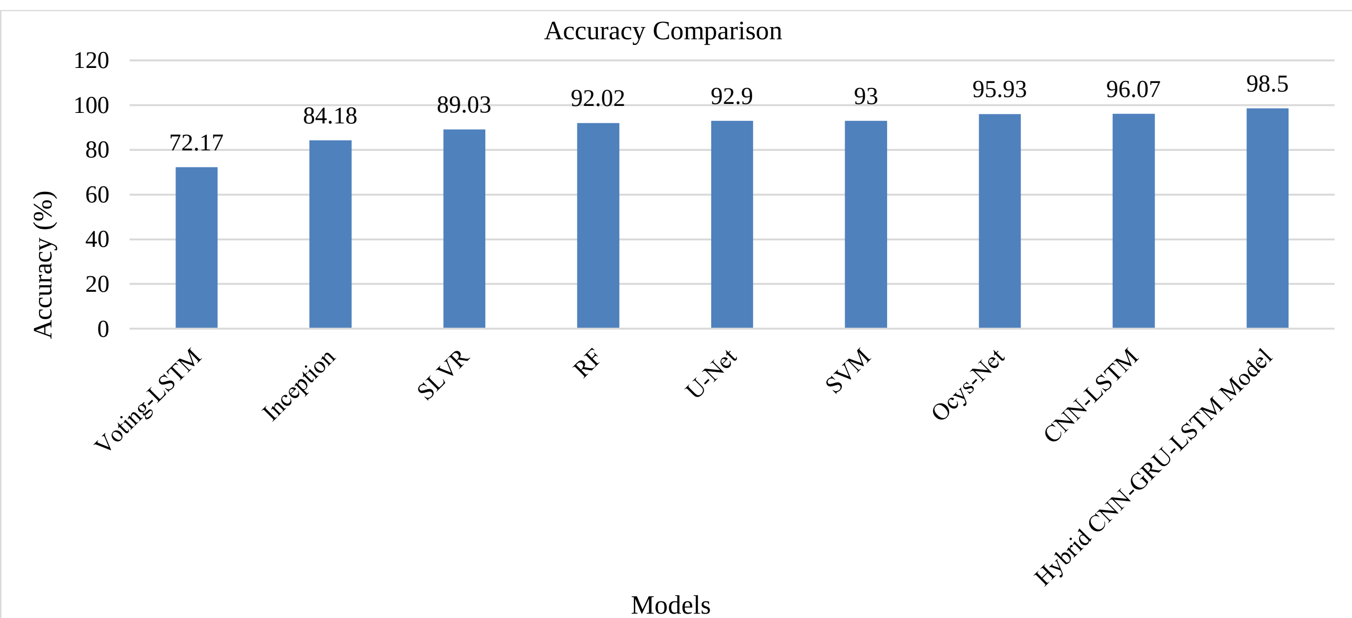


Fig. 12 Accuracy comparison of the proposed model with existing approaches

5. Conclusion

PCOS is a hormonal problem among women of reproductive age that is categorized by the existence of several small cysts on the ovaries, high levels of androgen and irregular menstrual periods. PCOS detection is the process of identifying and classifying the condition using medical imaging or clinical data. This study aims to create a robust and accurate approach DL approach for identifying PCOS from ultrasound images. The hybrid model proposed integrates CNN-GRU-LSTM models, each contributing its strengths in capturing spatial and temporal features of the data.

The dataset from Kaggle used for this research includes 3,856 ultrasound images, which are categorized into "infected" and "not infected" cases. Data preprocessing and augmentation methods such as resizing, normalization, and transformations were applied to enhance the variability and quality of the input data. The system architecture begins with CNN layers for feature extraction. Then, the GRU and LSTM layers are employed to capture sequential and long-term dependencies. The hybrid model concatenates the outputs from all three architectures to produce accurate predictions. The system achieves impressive performance, with 98.50% accuracy, 98.59% precision, 98.55% recall and a 98.48% F1 score, determining the efficiency of the hybrid model in accurately classifying PCOS from ultrasound images surpassing existing methods. While achieving high accuracy in PCOS detection, this study shows certain limitations. The model relies solely on ultrasound images, limiting its applicability to cases where imaging is unavailable. The real-time implementation in clinical settings requires further validation. Future research should explore multi-modal data integration and lightweight architectures optimized for real-time deployment.

Acknowledgements

I would like to convey my sincere thanks to all those who contributed to completing this research paper. I extend my heartfelt gratitude to my supervisor, family, colleagues and fellow researchers for their inspiration and understanding during the challenging phases of this work.

References

[1] Nithya Sathiadhas Puvaneswari, "*Detection of Polycystic Ovarian Syndrome using Convolutional Neural Network in Conjunction with Transfer Learning Models*," Master's Thesis, Dublin, National College of Ireland, 2022. [Google Scholar] [Publisher Link]

[2] Mohamed Abouhawwash et al., "Automatic Diagnosis of Polycystic Ovarian Syndrome Using Wrapper Methodology with Deep Learning Techniques," *Computer Systems Science and Engineering*, vol. 47, no. 1, pp. 239-253, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Sayma Alam Suha, and Muhammad Nazrul Islam, "An Extended Machine Learning Technique for Polycystic Ovary Syndrome Detection Using Ovary Ultrasound Image," *Scientific Reports*, vol. 12, no. 1, pp. 1-16, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Tahsin Istiyaq et al., "Polycystic Ovary Syndrome Detection Using Neural Network," Doctoral Dissertation, Brac University, pp. 1-34, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Zahra Zad et al., "Predicting Polycystic Ovary Syndrome with Machine Learning Algorithms from Electronic Health Records," *Frontiers in Endocrinology*, vol. 15, pp. 1-14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Jiekee Lim et al., "Machine Learning Classification of Polycystic Ovary Syndrome Based on Radial Pulse Wave Analysis," *BMC Complementary Medicine and Therapies*, vol. 23, no. 1, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Harshita Batra, and Leema Nelson, "DCADS: Data-Driven Computer Aided Diagnostic System using Machine Learning Techniques for Polycystic Ovary Syndrome," *International Journal of Performability Engineering*, vol. 19, no. 3, pp. 193-202, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Sayma Alam Suha, and Muhammad Nazrul Islam, "Exploring the Dominant Features and Data-Driven Detection of Polycystic Ovary Syndrome through Modified Stacking Ensemble Machine Learning Technique," *Heliyon*, vol. 9, no. 3, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Piyush Bhardwaj, and Parul Tiwari, "Manoeuvre of Machine Learning Algorithms in Healthcare Sector with Application to Polycystic Ovarian Syndrome Diagnosis," *Proceedings of Academia-Industry Consortium for Data Science,* Springer, Singapore, pp. 71-84, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] P. Rakshitha Kiran, and N.C. Naveen, "Op-RMSprop (Optimized-Root Mean Square Propagation) Classification for Prediction of Polycystic Ovary Syndrome (PCOS) Using Hybrid Machine Learning Technique," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 6, pp. 588-596, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] Pijush Dutta, Shobhandeb Paul, and Madhurima Majumder, "An Efficient SMOTE Based Machine Learning Classification for Prediction & Detection of PCOS," pp. 1-14, 2021. [Google Scholar]

[12] Subrato Bharati et al., "Ensemble Learning for Data-Driven Diagnosis of Polycystic Ovary Syndrome," *21st International Conference on Intelligent Systems Design and Applications,* Springer, Cham, pp. 1250-1259, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Neha A. Ranjith Kumar, and Vijayakumar Varadarajan, "Hybrid PCOS Net: A Synergistic CNN-LSTM Approach for Accurate Polycystic Ovary Syndrome Detection," pp. 1- 36, 2024. [Google Scholar]

[14] S. Sowmiya et al., "F-Net: Follicles Net an Efficient Tool for the Diagnosis of Polycystic Ovarian Syndrome Using Deep Learning Techniques," *PLOS one*, vol. 19, no. 8, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Abrar Alamoudi et al., "A Deep Learning Fusion Approach to Diagnosis the Polycystic Ovary Syndrome (PCOS)," *Applied Computational Intelligence and Soft Computing*, vol. 2023, no. 1, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] Junfang Fan et al., "Accurate Ovarian Cyst Classification with a Lightweight Deep Learning Model for Ultrasound Images," *IEEE Access*, vol. 11, pp. 110681-110691, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] Wenqi Lv et al., "Deep Learning Algorithm for Automated Detection of Polycystic Ovary Syndrome Using Scleral Images," *Frontiers in Endocrinology*, vol. 12, pp. 1-8, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] A.K.M. Salman Hosain, Md Humaion Kabir Mehedi, and Irteza Enan Kabir, "PCONET: A Convolutional Neural Network Architecture to Detect Polycystic Ovary Syndrome (PCOS) from Ovarian Ultrasound Images," IEEE *International Conference on Engineering and Emerging Technologies*, Kuala Lumpur, Malaysia, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Srivastava, S., Kumar, P., Chaudhry, V., & Singh, A. (2020). "Detection of Ovarian Cyst in Ultrasound Images Using Fine-Tuned VGG-16 Deep Learning Network," *SN Computer Science*, vol. 1, no. 2, pp. 1-8, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Anagha Choudhari, PCOS Detection Using Ultrasound Images, Kaggle.com, 2022. [Online] Available: https://www.kaggle.com/datasets/anaghachoudhari/pcos-detection-using-ultrasound-images/data

[21] Md Zahangir Alom et al., "A State-of-the-Art Survey on Deep Learning Theory and Architectures," *Electronics*, vol. 8, no. 3, pp. 1-66, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[22] Iram Bibi et al., "A Dynamic DL-Driven Architecture to Combat Sophisticated Android Malware," *IEEE Access*, vol. 8, pp. 129600-129612, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] Salma Alhagry, Aly Aly Fahmy, and Reda A. El-Khoribi, "Emotion Recognition based on EEG using LSTM Recurrent Neural Network," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, pp. 355-358, 2017. [CrossRef] [Google Scholar] [Publisher Link]