How to Cite:
Kaabra, R., & Rungta, S. (2022). Design of an intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning: ISM2SAITL. *International Journal of Health Sciences*, *6*(S6), 9742–9765. https://doi.org/10.53730/ijhs.v6nS6.12541

Design of an intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning: ISM2SAITL

Rachana Kaabra

Rungta College of Engineering and Technology, Bhilai, CG, India

*Corresponding author email: rachanakabra@yahoo.com

Sourabh Rungta

Rungta College of Engineering and Technology, Bhilai, CG, India

Email: sourabh@rungta.ac.in

***Abstract*---**Identification of medical diseases is a multidomain task that involves design of effective models for data collection, pre-processing, segmentation, feature extraction & reduction, with classification & post-processing. To design a medical expert system, researchers need to incorporate various input-specific processing models, that can cater to numerical, sequential, and image datasets. A wide variety of such system models are proposed by researchers for medical expert system design. Each of them utilizes some form of machine learning & deep learning interfaces. These interfaces assist in high-efficiency feature extraction, and disease classification via augmentation & interpolation of input datasets. But the scalability & performance of these models is limited due to their general-purpose architectural characteristics. Models that have high scalability, showcase moderate accuracy & precision performance when tested under multiple types of medical datasets. To improve this scalability, while maintaining high accuracy, precision & recall performance, a novel intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning is proposed in this text. The proposed ISM2SAITL model initially converts all input datasets into dynamically scaled & quantized 2D representations, which assists in deploying effective feature extraction methods. To extract these features, an ensemble feature evaluation engine is deployed, which assists in maximizing feature variance via hybrid swarm intelligence (HSI) model. The HSI model

uses a combination of particle swarm optimization (PSO) with elephant herding optimization (EHO) in order to extract disease-specific feature sets. These feature sets are classified using an ensemble of GoogLeNet, ResNet-18, and DenseNet models, which assists in high-accuracy feature classification. The results & patterns extracted from one type of disease are used to classify other disease types via a novel transfer learning model. The proposed ISM2SAITL model was evaluated for skin cancer, ophthalmic cancer, blood cancer, and Parkinson's disease datasets. These datasets had a variation in feature inputs,that included, clinical features, genomic features, and image features. These features were uniformly processed by the proposed model to achieve an accuracy of 98.9% across different datasets. Performance of the proposed ISM2SAITL model was compared w.r.t. various state-of-the-art approaches, and a precision improvement of 8%, recall improvement of 7.6%, and sensitivity improvement of 6.5% was observed. This performance enhancement was consistent across multiple disease types, which assists the model to achieve high scalability across different clinical applications.

Keywords---medical, expert, augmented, bioinspired, convolutional, neural, network, transfer, learning, feature, quantization, PSO, EHO, GoogLeNet, ResNet, DenseNet.

Introduction

Designing medical expert systems is a multidomain task that involves effective data collection, disease specific pre-processing, & segmentation, dataset specific feature extraction & feature selection, and application-specific classification & post-processing model designs. A typical medical expert system model is depicted in figure 1, wherein components like e-health sensors, clinical experts, patient information, and technologies are combined for final patient diagnosis [1].

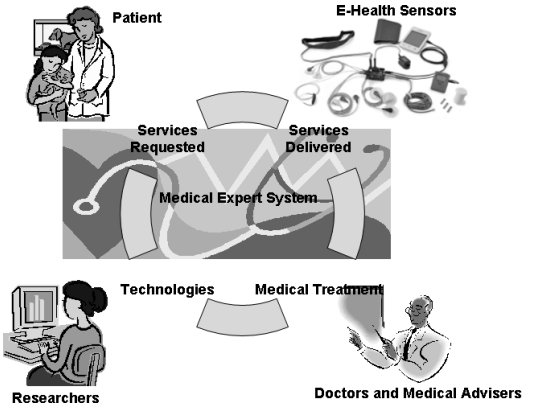

Figure 1. A typical medical expert system model

Based on this model, it can be observed that patients, e-health sensors, and clinical experts are non-tangible components, whose efficiency cannot be improved above a certain extent. Patients might give falsified information, sensors have quantization & other errors, while clinical experts always have diagnosis errors, which causes improper treatments. To reduce the probability of these errors, researchers develop high-efficiency classification modes, that can analyze patient & sensor inputs to simplify decision making at clinical-expert level. A review of these approaches [2, 3], along with their nuances, advantages, limitations & future research scopes is discussed in the next section of this text. From this review, it is observed that these models have moderate to high accuracy depending upon input dataset, which limits their scalability for real-time clinical applications. To improve this scalability, section 3 discusses design of an intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning.

This discussion proposes a dataset independent model for medical expert system design, and is capable of detecting multiple diseases with high precision, accuracy, recall & AUC performance. The proposed model uses incremental learning, which assists in continuous accuracy improvement, w.r.t. number of evaluations. The proposed model also uses transfer learning, which assists in adaptively transferring feature knowledge from one CNN model to other depending upon class types. Performance of the proposed model was compared with various state-of-the-art approaches in section 4, wherein accuracy, precision, recall, AUC and computation delay was evaluated. Based on this performance evaluation, it was observed that the proposed model is highly scalable, and has higher efficiency when compared with various state-of-the-art models. Finally, this text concludes with some interesting observations about the proposed model, and recommends various methods to further improve its performance.

Literature review

A wide variety of system models are deployed for clinical disease detection, and medical expert system designs. These models utilize multiple data representation modules, which assist in converting raw clinical readings into class-dependent feature vectors. For instance, work in [4, 5, 6] proposes design of heart disease prediction model using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) with augmented classifiers, short-term kidney disease classification via semi-supervised Multiple task learning, and long-term aggregation mechanism for better classification performance. These models showcase good performance for low-density datasets, but cannot be used for datasets with larger variance, due to which their scalability is limited. To overcome this limitation, work in [7] proposes a model for multiple support vector machines (MSVMs), which can be used to classify dense features without compromising classifier performance.

Extensions to this model are discussed in [8, 9, 10], wherein researchers have proposed use of hybrid random forest with a linear model (HRFLM), Conditional Random Field with Fully Convolutional Network (CRF FCN), & Genomic data use for classification of different diseases. These models are applied for disease specific datasets, but can be extended for multiple disease evaluations via

parameter tuning, & dataset-specific feature extractions. Such models are proposed in [11, 12, 13], wherein researchers have used Region of Interest based Transfer Learning with Convolutional Neural Network (CNN), Multiple view filtered Transfer Learning, and Multiple Task Deep Learning for skin diseases. Here, each of the reviewed model is trained for one type of disease, and then extended for other skin disease types via Transfer or Deep Learning Mechanisms, thus assisting in high scalability design of classifiers.

Models that utilize Deep Learning via CNNs outperform linear classification Methods, for instance, work in [14, 15, 16] proposes use of parallel cascades of Convolutional Neural Networks (PC CNN), Multiple Source Transfer Learning Via Multiple Kernel Support Vector Machine Plus (MSTL MKSVM), and multiple stratification models using CNN & its variants. These models assist in estimation of highly variant features, thereby improving overall classification performance, without compromising on delay & scalability performance. Performance of these models must be validated under different datasets, and can be extended via use of Dense Convolutional Neural Network [17], fuzzy logic-based classification [18], Deep Multiple instance Learning via Multiple Modalities [19], ensemble classification models [20], semi supervised Multitask Learning (SSMTL) [21], and Variance Maximized Deep Networks [22], which assist in improving feature variance for better signal representation under different datasets.

Specialized approaches that utilize Deep Learning Models for Keratoconus and Sub-Clinical Keratoconus Detection [23], Multivariate Analysis using CNNs & Decision Trees [24], Learning from Label Fuzzy Proportions (LLFP) [25], and Gaussian mixture models (GMM) with sensor CNN (SCNN) [26] are discussed which assist in improving correlative feature mapping between clinical & test features in order to improve overall classification performance. Extensions to these approaches are proposed in [27, 28, 29], wherein collaborative variational deep learning model (CVDL) with autoencoder, Deep CNN with photoplethysmography signals, and View classification CNN are proposed. These models are applicable to General purpose datasets, and can be extended via Q-Learning, Recurrent NN, and Long-Short-Term-Memory (LSTM) based models. Such extended models are discussed in [30, 31, 32], wherein Residual CNN, Robust sleep Network, and Graph Convolutional Networks are proposed. These models assist in extraction of high-density features, which assist in correlation-based matching of context-independent feature vectors. Thus, it can be observed that these Deep learning Models outperform linear classification models, but they are generic in nature, thus cannot be scaled for wider applications. To remove these limitations, next section proposes design of an intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning process. This model was evaluated on multiple datasets, and its performance was compared with various state-of-the-art techniques for validation of accuracy & scalability metrics under different scenarios.

Design of the proposed intrinsically scalable model for medical disease identification using hybrid swarm intelligence with augmented transfer learning

Based on the literature survey it can be observed that multiple models are proposed by researchers for identification of diseases based on scanned body parameters. Each of these models utilize machine learning & deep learning mechanisms to identify disease-specific feature patterns. The performance of deep learning models is limited due to use of general-purpose convolutions & classification layers. In contrast, models that use disease specific feature extraction & classification methods are highly specific to a particular disease type, which limits their scalability. To improve this scalability, while maintaining high accuracy, precision & recall performance, a novel intrinsic model for medical disease identification using hybrid swarm intelligence with augmented transfer learning is proposed in this text.

The proposed model is depicted in figure 2, and uses a combination of augmented feature extraction models, and uses domain specific hybrid swarm intelligence to extract highly variant feature sets. These feature sets are converted into 2D feature vectors, which are classified using an ensemble of GoogLeNet, ResNet-18, and DenseNet models. These models are selected due to their high-performance under separate datasets. For instance, GoogLeNet showcases better performance for skin cancer & and Parkinson's disease, while ResNet-18 performs better for ophthalmic cancer, and DenseNet model outperforms other models for blood cancer datasets. Design of the proposed model is segregated into multiple sub-parts, and is discussed in different sub-sections of this text. This will assist researchers to implement them in part(s) or as a whole depending upon their deployment requirements.

Design of the feature extraction layer

Input datasets are converted into multispectral features in order to enhance its representation capabilities. To perform this task, discrete wavelet features, discrete cosine transform (DCT) features, Fourier features, & iVector features are extracted. The wavelet features are able to identify entropy features, while DCT is able to reduce redundancy from features via converting them into frequency domain. Similarly, Fourier transform is used to extract these frequency components, while iVector is used for representing features into multispectral domains. To extract these features, different sets of identities are used, for instance, wavelet transform is evaluated via equation 1 & 2, wherein consecutive feature sets are utilized,

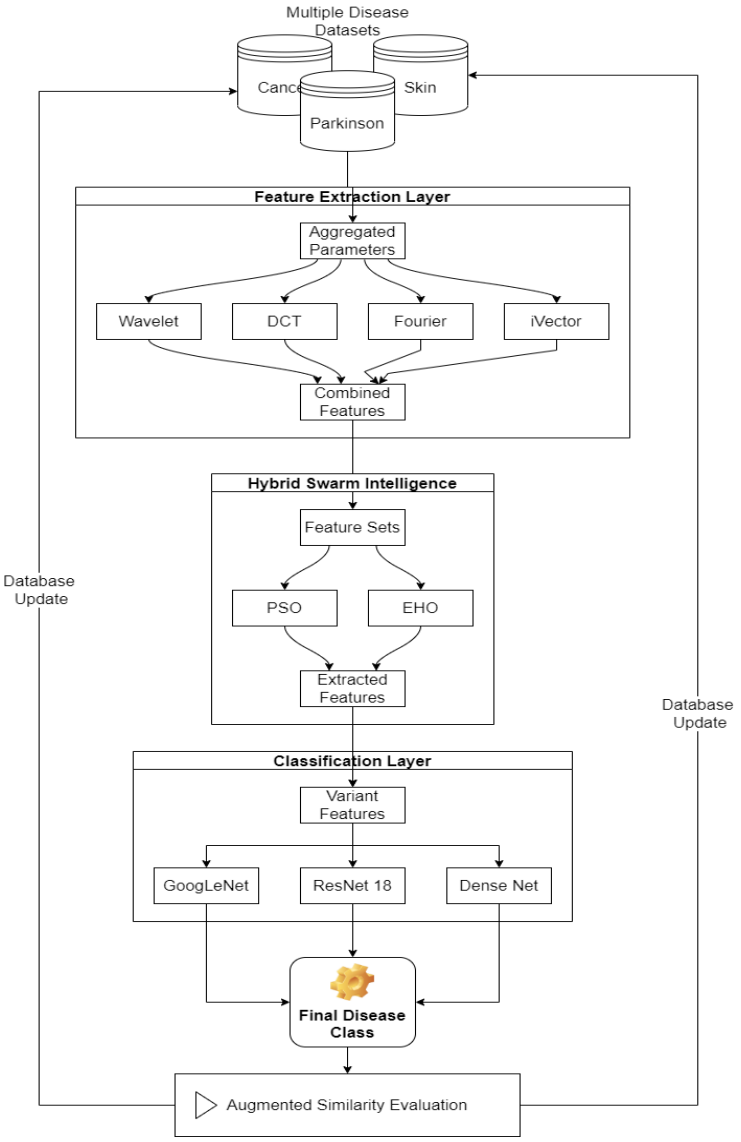

Figure 2. Overall flow of the proposed model

$$F(W_a) = \bigcup_{i=1}^{N_f-1} \frac{x_i + x_{i+1}}{2} \quad \ldots (1)$$

$$F(W_d) = \bigcup_{i=1}^{N_f-1} \frac{x_i - x_{i+1}}{2} \quad \ldots (2)$$

Where, W_a & W_d represents approximate & diagonal wavelet components, x represents dataset feature samples with N_f features. Similarly, DCT components are extracted via equation 3 as follows,

$$F(DCT_i) = \frac{1}{\sqrt{2 * N_f}} * x_i \sum_{j=1}^{N_f} x_j * \cos\left[\frac{\sqrt{-1} * (2 * i + 1) * \pi}{2 * N_f}\right] \quad \dots (3)$$

From these feature vectors, values which are above DCT threshold are used, while others are discarded from the feature set. This threshold is evaluated via equation 4 as follows,

$$DCT_{th} = \frac{\sum_{i=1}^{N_f} F(DCT_i) * L_{f_{DCT}}}{N_f} \quad \dots (4)$$

Where, $L_{f_{DCT}}$ is DCT learning factor, which is evaluated via the hybrid swarm intelligence layer described in section 3.2 of this text. The discrete Fourier transform (DFT) is evaluated via equation 5 as follows,

$$DFT_i = \sum_{j=1}^{N_f} x_j * \left[\cos\left(\frac{2 * \pi * i * j}{N_f}\right) - \sqrt{-1} * \sin\left(\frac{2 * \pi * i * j}{N_f}\right)\right] \quad \dots (5)$$

These features are evaluated to obtain multiple values of frequency component intensities. Some of these components have lower intensities, and thus introduce redundancies during classification. To remove these components, a Fourier threshold is evaluated via equation 6,

$$DFT_{th} = \sum_{i=1}^{N_f} DFT_i * \frac{L_{f_{DFT}}}{N_f} \quad \dots (6)$$

Where, $L_{f_{DFT}}$ represents DFT learning factor, and is tuned via the swarm intelligence layer as described in section 3.2 of this text. Both DCT & DFT are used for frequency level analysis, and are capable of extracting high intensity frequency patterns. To extend this analysis, iVectors are extracted via equation 7, which represents dataset features into multispectral scales.

$$F(iVector_i) = MAX\left(\bigcup_{j=1}^{N_f} x_j\right) + \begin{bmatrix} (1,1)_{var} & \cdots & (1,n)_{var} \\ \vdots & \ddots & \vdots \\ (n,1)_{var} & \cdots & (n,n)_{var} \end{bmatrix} * x_i \quad \dots (7)$$

Where, $(x, y)_{var}$ represents feature variance between samples $x \,\&\, y$, which is estimated via equation 8 as follows,

$$(x, y)_{var} = \frac{\exp\left(\frac{x^2}{2}\right)}{2 * pi * V(x) * V(y)} \quad \dots (8)$$

Where, $V(x)$ represents variance for the feature x, and assists in evaluation of feature consistency across different inputs. This variance is estimated via equation 9,

$$var(x) = \left(N_f - 1\right)^{-1} * \sum_{i=1}^{N_f} \left(x_i - \sum_{j=1}^{N_f} x_j * N_f^{-1} \right)^2 \quad \ldots (9)$$

Based on these identities, different features are extracted, which are capable of representing clinical observations into multispectral vectors. But due to use of multiple frequency components, these vectors contain feature-level redundancies which must be minimized for better classification performance. To reduce this redundancy, a hybrid swarm intelligence layer is deployed, which combines PSO & EHO models for highly variant feature selection. Design of this layer is described in the next sub-section of this text.

Design of hybrid swarm intelligence layer

The hybrid EHPSO model uses a combination of EHO & PSO for feature selection. As observed from figure 3, the model initially generates a set of Herds using EHO, and then optimizes each herd using PSO method.

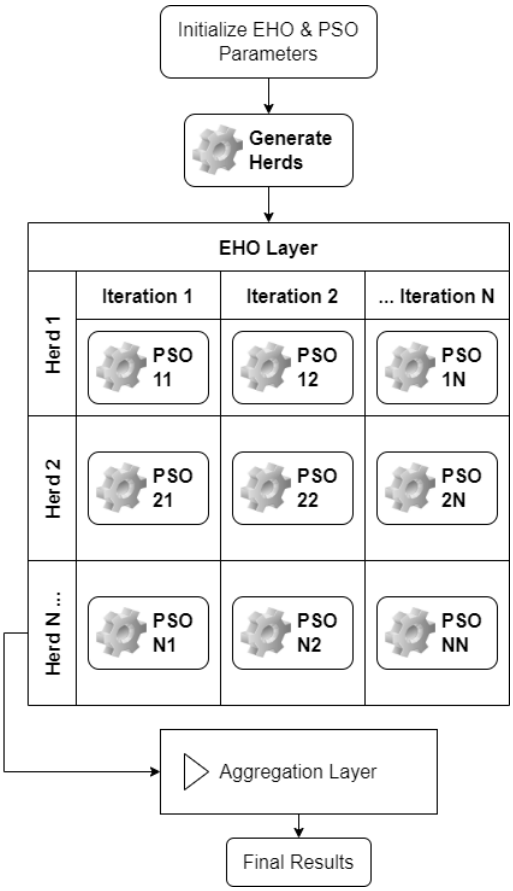

Figure 3. Overall flow of EHPSO Model

This model is capable of maximizing feature variance via continuous herd updates, and uses the following process to perform this task,

- Initialize EHPSO parameters,
- Number of herds (N_h)
- Number of iterations (N_i)
- Learning rate (L_r)
- Number of particles (N_p)
- Number of PSO iterations ($N_{i_{PSO}}$)
- Initially mark all herds as 'to be modified'
- For each iteration in 1 to N_i
- For each herd in 1 to N_h
  - If the herd is marked as 'not to be changed', then go to the next herd
  - Else, modify this herd via the following PSO process,
- Each herd is processed via PSO for continuous performance improvement.
- To perform this task, and Generate New particles for this herd, the following process is used,
- Generate stochastic values for DCT learning factor, DFT learning factor via equation 10,

$$L_{f_{DCT}} = STOCH(0.1, 1), L_{f_{DFT}} = STOCH(0.1,1) \quad \dots (10)$$

Where, $STOCH(x, y)$ represents a stochastic number between $x \,\&\, y$, which is generated via gold code random number generation process.

Select stochastic number of features from the input feature set via equation 11,

$$S_f = STOCH\left(L_r * N_f, N_f\right) \quad \dots (11)$$

Based on these features, evaluate particle velocity via equation 12,

$$V_i = \sqrt{\frac{\sum_{a=1}^{m}(F_a - \frac{\sum_{i=1}^{m}\sqrt{\frac{\sum_{j=1}^{n}(F_j - \frac{\sum_{l=1}^{n} F_l}{n})^2}{n-1}}}{m})^2}{m-1}} \quad \dots (12)$$

Where, m represents number of features in the current class, while n represents number of features in other classes.

Identify global best particle via equation 13,

$$GBest = V_i, where\ V_i = Max\left[\bigcup_{i=1}^{N_p} V_i\right] \quad \dots (13)$$

Assign current velocity of the particle as its present best position ($PBest$).
For each PSO iteration in 1 to $N_{i_{PSO}}$
Update particle velocity via equation 14,

$$New_{v_i} = r * V_i + C_{cog} * [V_i - PBest_i] + C_{soc} * [V_i - GBest] \quad \ldots(14)$$

Where, $r, C_{cog}, \& C_{soc}$ represents random number, cognitive learning factor, and social learning factor for the PSO model.

Based on this new velocity, features are randomly added or removed from the current particle. Select the particle with maximum velocity, and select it as current herd. Repeat this process for all modifiable herds, and assign PSO velocity as EHO fitness value via equation 15,

$$f_{EHO} = GBest \quad \ldots(15)$$

After each iteration, identify fitness threshold via equation 16,

$$f_{th} = \sum_{i=1}^{N_s} f_i * \frac{L_r}{N_s} \quad \ldots(16)$$

Herds with fitness lower than f_{th} represents features with lower variance, thus are discarded from the solution set by marking them as 'to be modified'. Remaining herds are marked as 'not to be modified'. This process is repeated for all iterations, and herd with maximum fitness is marked as 'Matriarch' herd. At the end of N_i iterations, herd with maximum fitness is selected, and its particles are used for final feature selection. To identify these features, select the particles with velocity more than threshold, which is evaluated via equation 17 as follows,

$$v_{th} = \sum_{i=1}^{N_p} v_i * \frac{L_r}{N_p} \quad \ldots(17)$$

Where, v_i represents particle velocity for particles with maximum variance. Based on this threshold, particles with velocity more than v_{th} are selected, and their features are evaluated via equation 18 as follows,

$$P_{sel} = Unique\left[\bigcup_{i=1}^{N_p} P[v_i > v_{th}]\right] \quad \ldots(18)$$

Where, $P[v_i]$ represents features for the particle with velocity v_i, while $Unique$ operator is used to reduce redundancy in the selected features. These features are converted into a standard 2D vector of size 224x224, which is compatible with CNN classification layers. To perform this conversion, following process is used,

- Initialize a 2D feature array (Arr_f), also initialize $r = 0, c = 0, i = 0$
- Assign values to array via equation 19,
- $Arr_f(r, c) = P_{set}(i) \quad \ldots(19)$
- Now increment, $i = i + 1, \& c = c + 1$
- If $c \geq 128, then\ r = r + 1, i = 0$

Based on this process, a 2D feature vector is generated, which assists in training the underling CNN classification model. Design of this model is discussed in the next section of this text.

Design of classification layer with augmented similarity evaluation for continuous database update

Once relevant features are extracted, and 2D feature vectors are obtained, then 3 different CNN models are trained to classify these vectors into 1 of N different classes. Due to the Generic Nature of feature extraction model, designs of these CNNs are used in their original forms, and utilize various convolutional, Maximum Variance Pooling, Dropout, and fully connected Neural Network (FCNN) layers. Standard configurations of GoogLeNet, ResNet-18, and DenseNet are depicted in figure 4 (a), 4 (b), and 4 (c) respectively, which are directly used without any modifications. Each of these models have common convolutional layers, which vary in terms of window sizes, stride sizes & padding sizes. This layer is used for extraction of augmented features from the 2D feature vector. These features are extracted via equation 20 as follows,

$$Conv_{out_{i,j}} = \sum_{a=-\frac{m}{2}}^{\frac{m}{2}} \sum_{b=-\frac{n}{2}}^{\frac{n}{2}} Arr_f(i-a, j-b) * ReLU\left(\frac{m+2a}{2}, \frac{n+2b}{2}\right) \quad \ldots(20)$$

Where, $ReLU$ represents feature activation rectilinear unit kernel, and is used to extract non-zero feature vectors, while m,n represents window sizes, and a,b represents stride & padding sizes respectively. These layers estimate a large number of features from the input set, which assist in design of a high-density feature classification process. The number of features can be estimated via equation 21,

$$F_{conv} = \frac{N_f + 2 * p - k + s}{s} \quad \ldots(21)$$

Where, p represents padding size, k represents window size of $ReLU$, and s represents stride size used for the given convolutional layer.

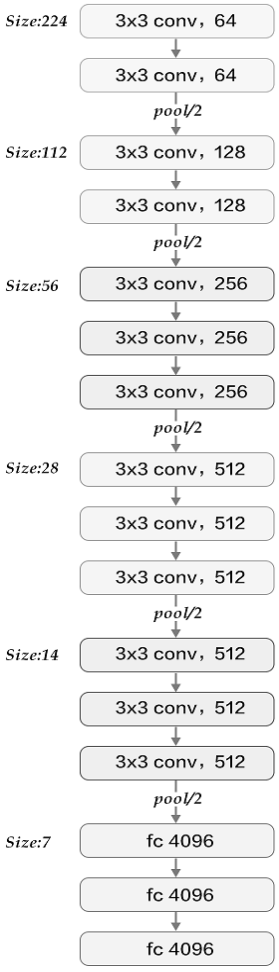

Figure 4 (a). The GoogLeNet model used for classification

As number of layers are increased, total number of features extracted via these layers is also increased. To reduce these features, a Maximum Pooling layer is used.

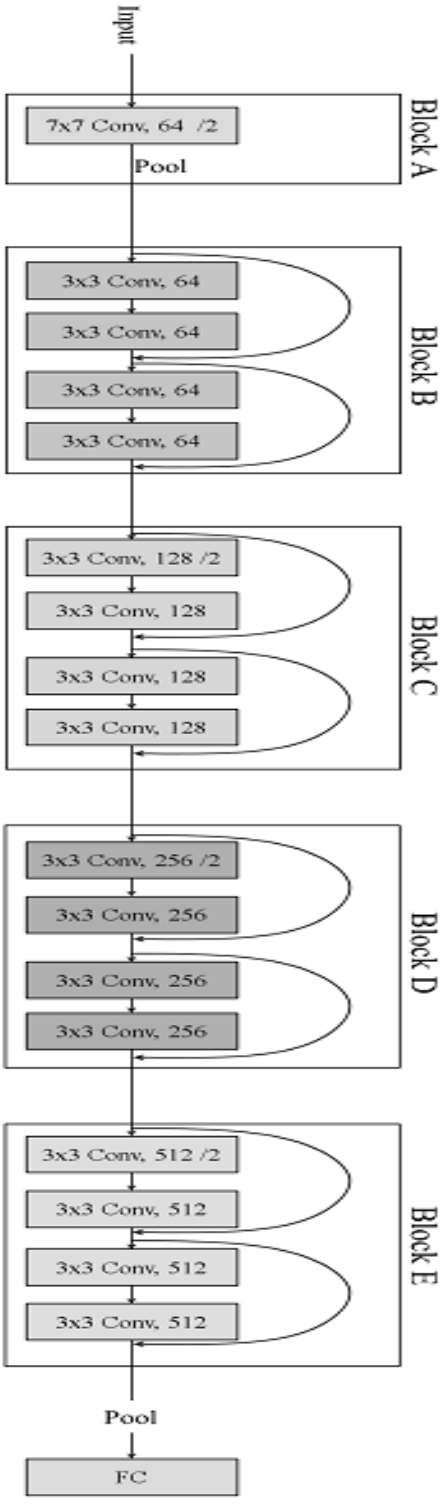

Figure 4 (b). The ResNet-18 model for classification

This layer estimates feature variance, and then evaluates a variance threshold in order to select most variant features.

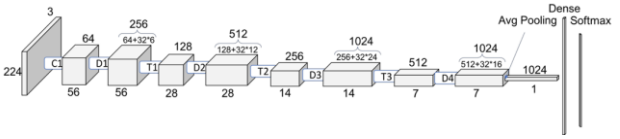

Figure 4 (c). The DenseNet model used for classification

This feature variance is estimated via equation 22,

$$var_{th} = \left(\frac{1}{f_{diff}} * \sum_{x \in N_f} Var_x \right)^{Var_x^{-1}} \quad \dots (22)$$

Where, f_{diff} & Var_x represents feature difference, and feature variance for the given convolutional layer. Features with variance more than var_{th} are selected, and passed to the next convolutional layer. Once the input vector is passed through all the layers, then a fully connected NN Model is used for final classification. This Model uses SoftMax activation for classifying the input vector into 1 of N classes as observed from equation 24,

$$c_{out} = SoftMax \left(\sum_{i=1}^{N_f} f_i * w_i + b \right) \quad \dots (24)$$

Where, f_i represents convolutional features from the final layer, w_i represents their weights, b is a tuned bias value, while c_{out} represents output disease class. Based on this evaluation, input feature vector is categorized into 1 of N different classes. This process is repeated for GoogLeNet, ResNet & DenseNet classifiers, and the final output (f_{out}) class is evaluated via equation 25,

$$f_{out} = MODE(GoogLe_{cout}, Res_{cout}, Dense_{cout}) \quad \dots (25)$$

After classification, the feature vector is compared with training dataset via cosine similarity metric for correlation checks. The cosine similarity metric is evaluated from equation 25,

$$SIM(F_{test}, F_{train}) = \frac{\sum_{i=1}^{N_f} F_{test_i} * F_{train_i}}{\sum_{i=1}^{N_f} \sqrt{F_{test_i}^2} * \sum_{i=1}^{N_f} \sqrt{F_{train_i}^2}} \quad \dots (26)$$

If this similarity value is more than 0.999, then the test input is added to a retraining dataset with respective tag, else it is removed from the testing set. Once number of entries in the retraining set are more than $L_r * Len(Set_{train})$, then the model is retrained for continuous performance improvement. This performance is validated in multiple datasets, and is compared with various state-of-the-art models in the next section of this text.

Result analysis and comparison

The proposed ISM2SAITL model uses a combination of bioinspired computing with augmented feature extraction & deep convolutional Neural Networks (CNNs) in order to identify different disease classes. To validate performance of the proposed model, the following datasets were used,

- MNIST: HAM10000 dataset for Skin cancer, which is available at, https://www.kaggle.com/kmader/skin-cancer-mnist-ham10000
- Aberrant epithelial-mesenchymal transition (EMT) dataset for Ophthalmic cancer, which is available at, http://biogps.org/dataset/E-GEOD-15205/tgf-or-tnf-time-series-in-arpe19/
- Blood Cancer Diagnosis dataset, available at, https://www.kaggle.com/sarques/blood-cancer-diagnosis/data
- Parkinson's Data Set, available at https://archive.ics.uci.edu/ml/datasets/parkinsons

These datasets were combined to form a set of 48k tagged records, which vary in terms of number of features, feature type, and intensity of features. The dataset was split in 70:30 ratio, wherein 70% entries were used for training, while remaining 30% were used for testing the classifier performance. The combined dataset was evaluated in terms of accuracy (A), precision (P), recall (R), & sensitivity (S) metrics, and compared with CRF FCN [9], SCNN [26], and CVDL [27] models. Based on this comparison, researchers will be able to identify performance patterns of the proposed model when compared with standard classification approaches. Based on these conditions, accuracy of classification is tabulated in table 1 & figure 5 as follows,

Table 1
Accuracy of classification for different datasets

Number of test samples	A (%) CRF FCN [9]	A (%) SCNN [26]	A (%) CVDL [27]	A (%) ISM2 SAITL
1000	85.60	86.20	88.21	92.57
1500	85.90	86.30	88.44	93.00
2000	86.20	86.50	88.85	93.79
2500	86.90	86.90	89.85	94.60
3000	88.50	88.10	90.74	95.28
3500	89.10	88.20	91.10	95.62
4000	89.50	88.50	91.41	95.92
4500	89.60	88.90	91.67	96.23
5000	89.90	89.10	91.97	96.56
5500	90.40	89.30	92.31	96.94
6000	90.70	89.60	92.67	97.31
6500	90.90	90.20	93.05	97.63
7000	91.20	90.60	93.33	97.87
7500	91.30	90.90	93.51	98.04
8000	91.40	91.10	93.67	98.18
8500	91.50	91.30	93.77	98.35

9000	91.50	91.40	93.95	98.54
9500	91.60	91.90	94.18	98.74
10000	91.60	92.20	94.33	98.88
11000	91.60	92.50	94.46	99.00
12000	91.70	92.60	94.56	99.09
13000	91.70	92.80	94.64	99.20
14000	91.70	92.90	94.74	99.31
14500	91.80	93.10	94.86	99.42

From this comparison & figure 5, it can be observed that the proposed model is 8.5% better than CRF FCN [9], 6.3% better than SCNN [26], and 4.6% better than CVDL [27], which is due to use of different CNN models, which assist in improving classification performance for different datasets. Moreover, the proposed model utilizes various feature selection & variance maximization techniques, which assists in improving classification performance. This makes the model applicable for various real-time clinical applications. Similarly, precision of different models is tabulated in table 2 as follows,

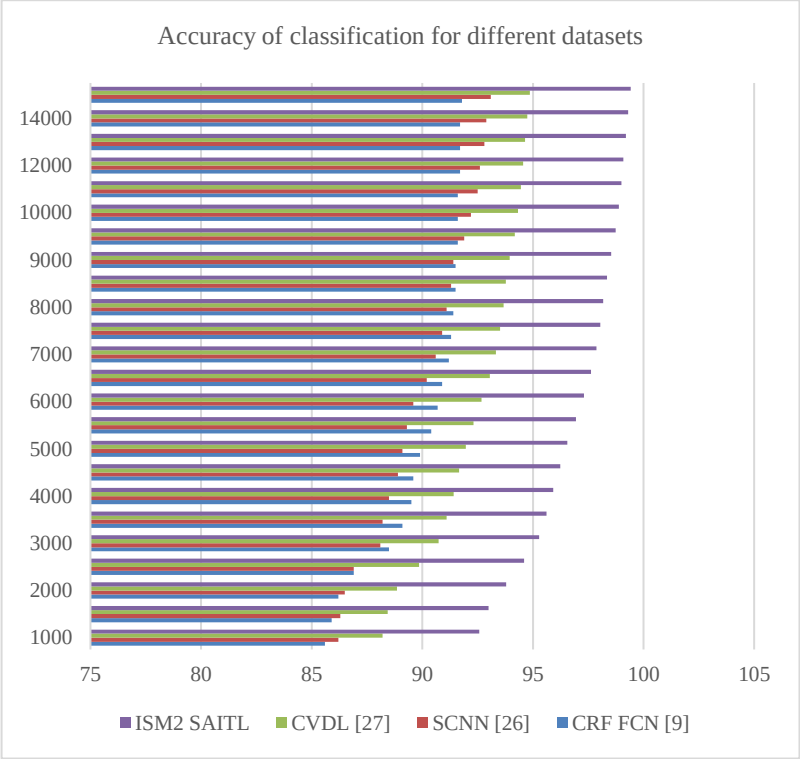

Figure 5. Accuracy of classification for different datasets

Table 2
Precision of classification for different datasets

Number of test samples	P (%) CRF FCN [9]	P (%) SCNN [26]	P (%) CVDL [27]	P (%) ISM2 SAITL
1000	81.81	78.50	88.53	90.91
1500	82.29	78.73	89.14	91.51
2000	83.19	79.36	89.79	92.23
2500	83.81	79.59	90.47	92.79
3000	84.76	80.27	91.08	93.27
3500	85.10	80.50	91.38	93.59
4000	85.43	80.73	91.69	93.90
4500	85.71	81.00	91.99	94.23
5000	86.00	81.23	92.32	94.57
5500	86.33	81.59	92.68	94.91
6000	86.62	81.91	93.00	95.21
6500	86.76	82.32	93.28	95.45
7000	86.95	82.59	93.50	95.63
7500	87.05	82.82	93.64	95.80
8000	87.10	82.95	93.81	95.96
8500	87.19	83.27	93.97	96.15
9000	87.19	83.45	94.14	96.30
9500	87.24	83.82	94.32	96.46
10000	87.29	84.00	94.45	96.57
11000	87.29	84.23	94.55	96.68
12000	87.33	84.32	94.65	96.78
13000	87.38	84.50	94.75	96.89
14000	87.40	84.61	94.85	96.99
14500	87.47	84.77	94.96	97.10

From this comparison & figure 6, it can be observed that the proposed model has 9.4% better precision than CRF FCN [9], 12.5% better precision than SCNN [26], and 2.3% better precision than CVDL [27], which is due to use of different CNN classifiers for each disease type.

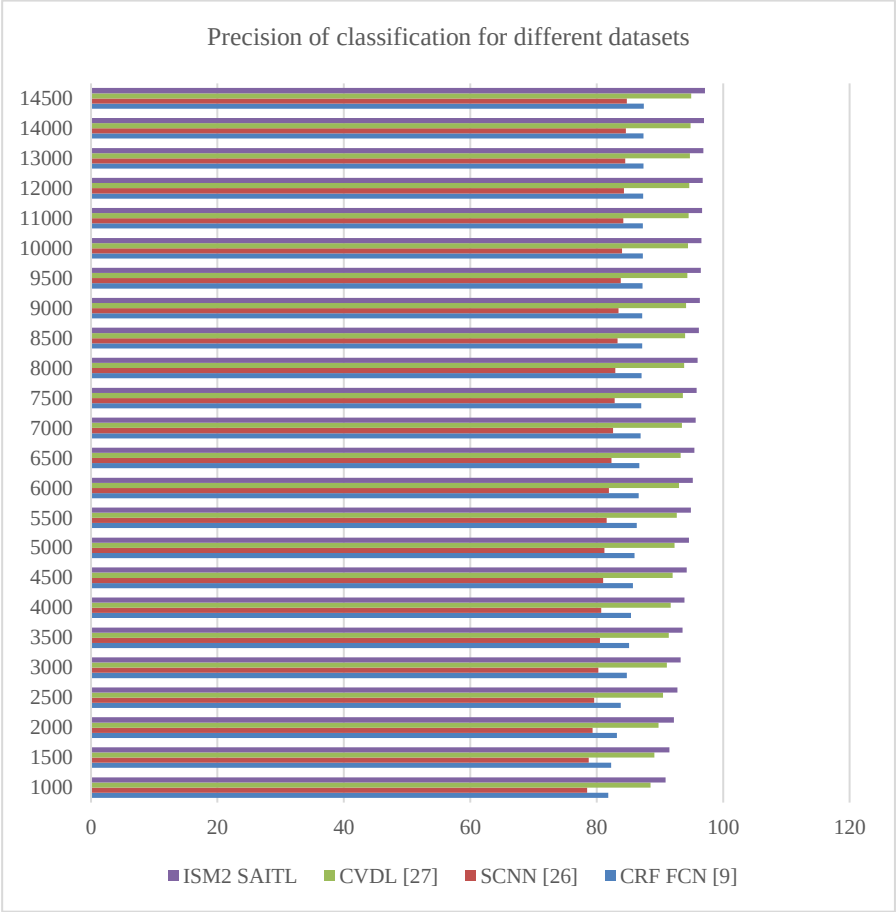

Figure 6. Precision of classification for different datasets

Furthermore, the proposed model utilizes various feature selection & variance maximization techniques, which assists in improving classification performance. This makes the model applicable for multiple real-time clinical applications. Similarly, recall of different models is tabulated in table 3 as follows,

Table 3
Recall of classification for different datasets

Number of test samples	R (%) CRF FCN [9]	R (%) SCNN [26]	R (%) CVDL [27]	R (%) ISM2 SAITL
1000	83.90	82.43	88.58	92.00
1500	84.39	82.72	89.05	92.63
2000	85.03	83.09	89.74	93.35
2500	85.99	83.72	90.54	93.99
3000	86.86	84.27	91.08	94.44
3500	87.28	84.48	91.40	94.76
4000	87.56	84.78	91.69	95.07
4500	87.80	85.06	91.99	95.40

5000	88.16	85.30	92.32	95.75
5500	88.51	85.60	92.66	96.09
6000	88.75	86.01	93.00	96.40
6500	88.95	86.43	93.29	96.65
7000	89.13	86.73	93.50	96.84
7500	89.21	86.94	93.66	96.99
8000	89.30	87.16	93.80	97.16
8500	89.35	87.36	93.96	97.34
9000	89.38	87.64	94.15	97.51
9500	89.43	87.98	94.32	97.67
10000	89.44	88.23	94.45	97.79
11000	89.48	88.41	94.56	97.89
12000	89.53	88.55	94.65	97.99
13000	89.54	88.70	94.75	98.10
14000	89.59	88.84	94.85	98.20
14500	89.65	89.00	94.96	96.94

From this comparison & figure 7, it can be observed that the proposed model has 6.8% better recall than CRF FCN [9], 16.2% better recall than SCNN [26], and 1.9% better recall than CVDL [27], which is due to use of feature augmentation & selection processes.

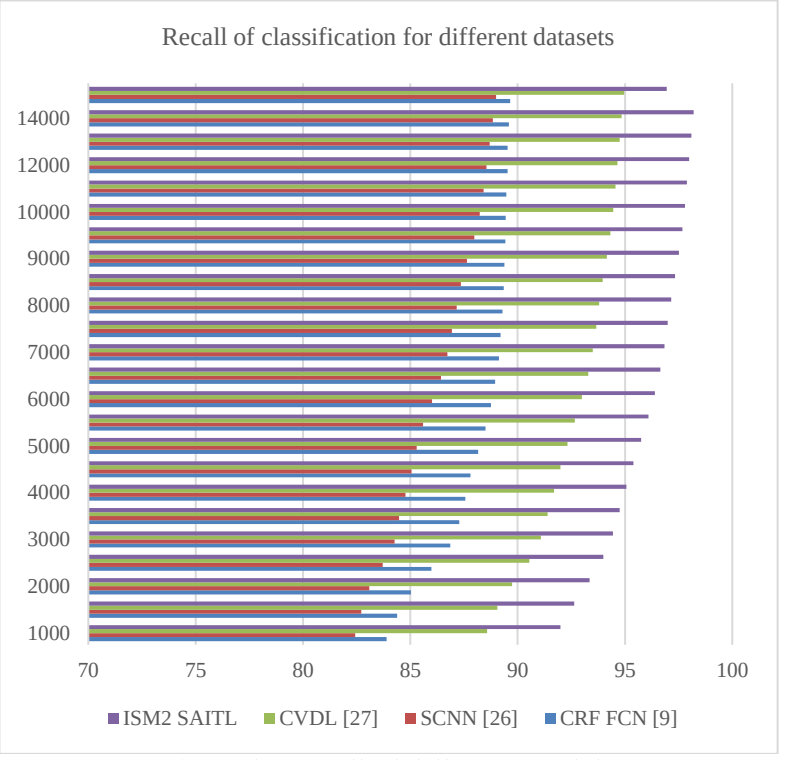

Figure 7. Recall of different models

Furthermore, the proposed model utilizes various classification models for different datasets, which assists in improving recall performance. This makes the model applicable for multiple real-time clinical applications. Similarly, sensitivity of different models is tabulated in table 4 as follows,

Table 4
Sensitivity of classification for different datasets

Number of test samples	S (%) CRF FCN [9]	S (%) SCNN [26]	S (%) CVDL [27]	S (%) ISM2 SAITL
1000	83.77	79.72	85.58	88.86
1500	84.19	79.92	86.01	89.40
2000	84.81	80.31	86.57	90.12
2500	85.57	80.71	87.37	90.77
3000	86.71	81.50	88.03	91.29
3500	87.16	81.67	88.35	91.60
4000	87.50	81.94	88.64	91.90
4500	87.71	82.24	88.92	92.21
5000	88.02	82.46	89.23	92.54
5500	88.42	82.74	89.56	92.88
6000	88.69	83.07	89.89	93.20
6500	88.87	83.53	90.20	93.46
7000	89.09	83.84	90.43	93.66
7500	89.19	84.08	90.58	93.82
8000	89.26	84.26	90.74	93.97
8500	89.35	84.49	90.87	94.14
9000	89.36	84.68	91.04	94.31
9500	89.42	85.06	91.23	94.47
10000	89.44	85.30	91.36	94.59
11000	89.46	85.53	91.47	94.70
12000	89.52	85.64	91.57	94.80
13000	89.54	85.81	91.66	94.90
14000	89.56	85.92	91.76	95.00
14500	89.64	86.08	91.87	94.66

From this comparison & figure 8, it can be observed that the proposed model has 4.9% more sensitive than CRF FCN [9], 7.6% more sensitive than SCNN [26], and 2.5% more sensitive than CVDL [27], which is due to use of different CNN classifiers for each disease type. Furthermore, the proposed model utilizes various feature selection & variance maximization techniques, which assists in improving classification performance.

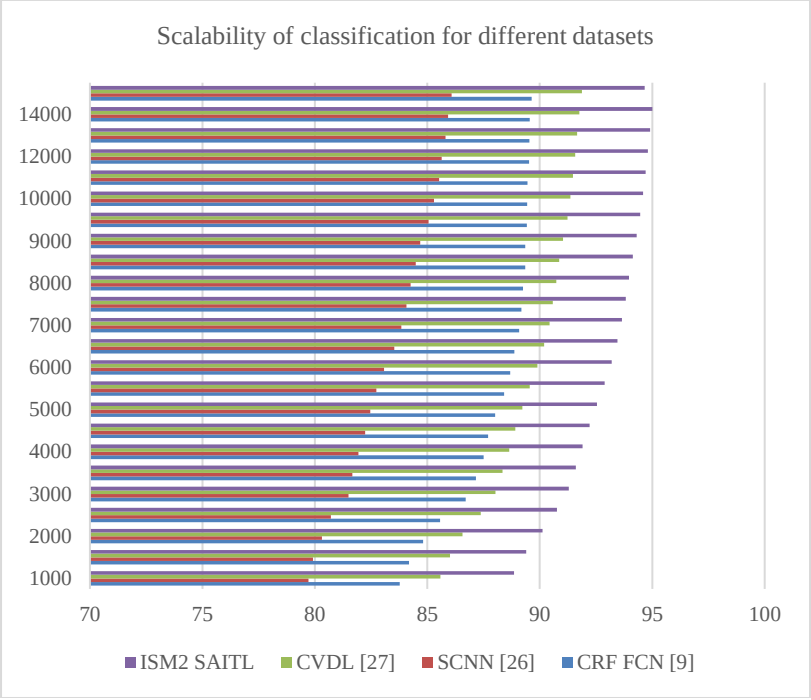

Figure 8. Scalability of classification for different datasets

Due to such a wide improvement in performance, the proposed model is applicable for clinical deployments, that require high-accuracy, better precision, higher recall, and more sensitivity for different datasets & different application scenarios.

Conclusion and future work

The proposed ISM2SAITL model uses a combination of various deep learning classification methods for improving disease detection performance. These classifiers are combined with a general-purpose high-density feature extraction unit, which utilizes wavelet, cosine, Fourier, and iVector feature sets. These feature sets are processed via an intelligent Hybrid EHPSO model, that integrates variance maximization-based fitness evaluation for selection of highly variant features. The extracted features are converted into 2D arrays for better convolutional processing capabilities. Due to use of such intelligent techniques, the proposed model is capable of achieving an accuracy of 98.72% when averaged across multiple feature sets, this performance is 8.5% better than CRF FCN [9], 6.3% better than SCNN [26], and 4.6% better than CVDL [27], which makes the model highly applicable for a wide variety of deployment scenarios. Similarly, the proposed model is observed to have higher precision, better recall, and better sensitivity when compared with various state-of-the-art Methods.

Due to this improvement in performance, the proposed model is useful for real-time clinical applications. Moreover, the model uses a correlation-based matching framework, which modifies its internal training metrics and assists in incremental accuracy improvement. Thus, the model's performance improves w.r.t. number of

clinical evaluations, thereby making it highly applicable & versatile for real-time applications. In future, researchers can integrate Q-Learning, Deep Reinforcement Learning (DRL), Recurrent Neural Networks (RNNs), etc. to further improve classification performance. Researchers can also validate performance of the proposed model for different datasets, which will assist in estimating any lacunas in the current research, which can assist in modelling better techniques for context-sensitive application deployments.

References

1. A. A. Adegun and S. Viriri, "FCN-Based DenseNet Framework for Automated Detection and Classification of Skin Lesions in Dermoscopy Images," in IEEE Access, vol. 8, pp. 150377-150396, 2020, doi: 10.1109/ACCESS.2020.3016651.
2. A. Bizzego et al., "Integrating deep and radiomics features in cancer bioimaging," 2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2019, pp. 1-8, doi: 10.1109/CIBCB.2019.8791473.
3. A. Guillot and V. Thorey, "RobustSleepNet: Transfer Learning for Automated Sleep Staging at Scale," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 29, pp. 1441-1451, 2021, doi: 10.1109/TNSRE.2021.3098968.
4. D. Chicco and G. Jurman, "Arterial Disease Computational Prediction and Health Record Feature Ranking Among Patients Diagnosed With Inflammatory Bowel Disease," in IEEE Access, vol. 9, pp. 78648-78657, 2021, doi: 10.1109/ACCESS.2021.3084063.
5. D. Kumar et al., "Automatic Detection of White Blood Cancer From Bone Marrow Microscopic Images Using Convolutional Neural Networks," in IEEE Access, vol. 8, pp. 142521-142531, 2020, doi: 10.1109/ACCESS.2020.3012292.
6. E. Iadanza et al., "Automatic Detection of Genetic Diseases in Pediatric Age Using Pupillometry," in IEEE Access, vol. 8, pp. 34949-34961, 2020, doi: 10.1109/ACCESS.2020.2973747.
7. E. Smistad et al., "Real-Time Automatic Ejection Fraction and Foreshortening Detection Using Deep Learning," in IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 67, no. 12, pp. 2595-2604, Dec. 2020, doi: 10.1109/TUFFC.2020.2981037.
8. H. Wang, Z. Huang, D. Zhang, J. Arief, T. Lyu and J. Tian, "Integrating Co-Clustering and Interpretable Machine Learning for the Prediction of Intravenous Immunoglobulin Resistance in Kawasaki Disease," in IEEE Access, vol. 8, pp. 97064-97071, 2020, doi: 10.1109/ACCESS.2020.2996302.
9. H. Zhang et al., "Multi-Source Transfer Learning Via Multi-Kernel Support Vector Machine Plus for B-Mode Ultrasound-Based Computer-Aided Diagnosis of Liver Cancers," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 10, pp. 3874-3885, Oct. 2021, doi: 10.1109/JBHI.2021.3073812.
10. J. Bian, S. Zhang, S. Wang, J. Zhang and J. Guo, "Skin Lesion Classification by Multi-View Filtered Transfer Learning," in IEEE Access, vol. 9, pp. 66052-66061, 2021, doi: 10.1109/ACCESS.2021.3076533.

11. J. M. Gálvez et al., "Towards Improving Skin Cancer Diagnosis by Integrating Microarray and RNA-Seq Datasets," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 7, pp. 2119-2130, July 2020, doi: 10.1109/JBHI.2019.2953978.
12. J. Yang et al., "Self-Paced Balance Learning for Clinical Skin Disease Recognition," in IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 8, pp. 2832-2846, Aug. 2020, doi: 10.1109/TNNLS.2019.2917524.
13. Joshi, N. R., & Sebastian, R. (2022). Effect of medical traveller's satisfaction on their intention to revisit for medical tourism in India. International Journal of Health Sciences, 6(S5), 5007–5017. https://doi.org/10.53730/ijhs.v6nS5.9701
14. L. Song, J. Lin, Z. J. Wang and H. Wang, "An End-to-End Multi-Task Deep Learning Framework for Skin Lesion Analysis," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 10, pp. 2912-2921, Oct. 2020, doi: 10.1109/JBHI.2020.2973614.
15. M. Bernardini, L. Romeo, E. Frontoni and M. -R. Amini, "A Semi-Supervised Multi-Task Learning Approach for Predicting Short-Term Kidney Disease Evolution," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 10, pp. 3983-3994, Oct. 2021, doi: 10.1109/JBHI.2021.3074206.
16. M. Sahasrabudhe et al., "Deep Multi-Instance Learning Using Multi-Modal Data for Diagnosis of Lymphocytosis," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 6, pp. 2125-2136, June 2021, doi: 10.1109/JBHI.2020.3038889.
17. Muhammad, L.J., Algehyne, E.A. Fuzzy based expert system for diagnosis of coronary artery disease in nigeria. Health Technol. 11, 319–329 (2021). https://doi.org/10.1007/s12553-021-00531-z.
18. N. L. Fitriyani, M. Syafrudin, G. Alfian and J. Rhee, "HDPM: An Effective Heart Disease Prediction Model for a Clinical Decision Support System," in IEEE Access, vol. 8, pp. 133034-133050, 2020, doi: 10.1109/ACCESS.2020.3010511.
19. O. Zhang et al., "Explainability Metrics of Deep Convolutional Networks for Photoplethysmography Quality Assessment," in IEEE Access, vol. 9, pp. 29736-29745, 2021, doi: 10.1109/ACCESS.2021.3054613.
20. P. Bizopoulos and D. Koutsouris, "Deep Learning in Cardiology," in IEEE Reviews in Biomedical Engineering, vol. 12, pp. 168-193, 2019, doi: 10.1109/RBME.2018.2885714.
21. Q. Ye et al., "Method of Tumor Pathological Micronecrosis Quantification Via Deep Learning From Label Fuzzy Proportions," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 9, pp. 3288-3299, Sept. 2021, doi: 10.1109/JBHI.2021.3071276.
22. R. Ashraf et al., "Region-of-Interest Based Transfer Learning Assisted Framework for Skin Cancer Detection," in IEEE Access, vol. 8, pp. 147858-147871, 2020, doi: 10.1109/ACCESS.2020.3014701.
23. R. Feng et al., "KerNet: A Novel Deep Learning Approach for Keratoconus and Sub-Clinical Keratoconus Detection Based on Raw Data of the Pentacam HR System," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 10, pp. 3898-3910, Oct. 2021, doi: 10.1109/JBHI.2021.3079430.
24. S. Chambon, M. N. Galtier, P. J. Arnal, G. Wainrib and A. Gramfort, "A Deep Learning Architecture for Temporal Sleep Stage Classification Using

Multivariate and Multimodal Time Series," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 4, pp. 758-769, April 2018, doi: 10.1109/TNSRE.2018.2813138.

25. S. Chi, Y. Tian, F. Wang, Y. Wang, M. Chen and J. Li, "Deep Semisupervised Multitask Learning Model and Its Interpretability for Survival Analysis," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 8, pp. 3185-3196, Aug. 2021, doi: 10.1109/JBHI.2021.3064696.
26. S. Dong, Q. Yang, Y. Fu, M. Tian and C. Zhuo, "RCoNet: Deformable Mutual Information Maximization and High-Order Uncertainty-Aware Learning for Robust COVID-19 Detection," in IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 8, pp. 3401-3411, Aug. 2021, doi: 10.1109/TNNLS.2021.3086570.
27. S. Khan, M. Sajjad, T. Hussain, A. Ullah and A. S. Imran, "A Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images," in IEEE Access, vol. 9, pp. 10657-10673, 2021, doi: 10.1109/ACCESS.2020.3048172.
28. S. Mohan, C. Thirumalai and G. Srivastava, "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques," in IEEE Access, vol. 7, pp. 81542-81554, 2019, doi: 10.1109/ACCESS.2019.2923707.
29. Suryasa, I. W., Rodríguez-Gámez, M., & Koldoris, T. (2021). Health and treatment of diabetes mellitus. *International Journal of Health Sciences*, *5*(1), i-v. https://doi.org/10.53730/ijhs.v5n1.2864
30. Suryatika, I. B. M., Poniman, S., Manuaba, I. B. P., Yasa, I. W. P. S., & Sutapa, G. N. (2021). The potential cancer risk on body organs as abdomen CT-scan exposure result. International Journal of Life Sciences, 5(3), 171–178. https://doi.org/10.53730/ijls.v5n3.1721
31. T. Gangavarapu, G. S. Krishnan, S. K. S and J. Jeganathan, "FarSight: Long-Term Disease Prediction Using Unstructured Clinical Nursing Notes," in IEEE Transactions on Emerging Topics in Computing, vol. 9, no. 3, pp. 1151-1169, 1 July-Sept. 2021, doi: 10.1109/TETC.2020.2975251.
32. W. Zhang, C. Su and C. He, "Rehabilitation Exercise Recognition and Evaluation Based on Smart Sensors With Deep Learning Framework," in IEEE Access, vol. 8, pp. 77561-77571, 2020, doi: 10.1109/ACCESS.2020.2989128.
33. X. Deng and F. Huangfu, "Collaborative Variational Deep Learning for Healthcare Recommendation," in IEEE Access, vol. 7, pp. 55679-55688, 2019, doi: 10.1109/ACCESS.2019.2913468.
34. Y. Ren, Y. Shi, K. Zhang, Z. Chen and Z. Yan, "Medical Treatment Migration Prediction Based on GCN via Medical Insurance Data," in IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 9, pp. 2516-2522, Sept. 2020, doi: 10.1109/JBHI.2020.3008493.
35. Y. Zhang, X. Wang, N. Han and R. Zhao, "Ensemble Learning Based Postpartum Hemorrhage Diagnosis for 5G Remote Healthcare," in IEEE Access, vol. 9, pp. 18538-18548, 2021, doi: 10.1109/ACCESS.2021.3051215.