SN Computer Science (2021) 2:369
https://doi.org/10.1007/s42979-021-00741-2

ORIGINAL RESEARCH

Cervical Cytology Classification Using PCA and GWO Enhanced Deep Features Selection

Hritam Basak[1] · **Rohit Kundu**[1] · **Sukanta Chakraborty**[2] · **Nibaran Das**[3]

Received: 13 February 2021 / Accepted: 8 June 2021 / Published online: 7 July 2021

Abstract

Cervical cancer is one of the most deadly and common diseases among women worldwide. It is completely curable if diagnosed in an early stage, but the tedious and costly detection procedure makes it unviable to conduct population-wise screening. Thus, to augment the effort of the clinicians, in this paper, we propose a fully automated framework that utilizes deep learning and feature selection using evolutionary optimization for cytology image classification. The proposed framework extracts deep feature from several convolution neural network (CNN) models and uses a two-step feature reduction approach to ensure reduction in computation cost and faster convergence. The features extracted from the CNN models form a large feature space whose dimensionality is reduced using principal component analysis while preserving 99% of the variance. A non-redundant, optimal feature subset is selected from this feature space using an evolutionary optimization algorithm, the grey wolf optimizer, thus improving the classification performance. Finally, the selected feature subset is used to train an support vector machine classifier for generating the final predictions. The proposed framework is evaluated on three publicly available benchmark datasets: Mendeley Liquid Based Cytology (4-class) dataset, Herlev Pap Smear (7-class) dataset, and the SIPaKMeD Pap Smear (5-class) dataset achieving classification accuracies of 99.47, 98.32 and 97.87%, respectively, thus justifying the reliability of the approach. The relevant codes for the proposed approach can be found in: https://github.com/DVLP-CMATERJU/Two-Step-Feature-Enhancement.

Keywords Deep learning · Cervical cytology · Evolutionary optimization · Principal component analysis · Grey wolf optimization

This article is part of the topical collection "AI and Deep Learning Trends in Healthcare" guest edited by KC Santosh, Paolo Soda and Zalelam Temesgen.

✉ Nibaran Das
nibaran.das@jadavpuruniversity.in

Hritam Basak
hritambasak48@gmail.com

Rohit Kundu
rohitkunduju@gmail.com

Sukanta Chakraborty
drsukantachakraborty@gmail.com

[1] Department of Electrical Engineering, Jadavpur University, 188, Raja S.C. Mullick Road, Jadavpur, Kolkata, West Bengal 700032, India

[2] Theism Medical Diagnostics Centre, Dum Dum, Kolkata, West Bengal 700030, India

[3] Department of Computer Science and Engineering, Jadavpur University, 188, Raja S.C. Mullick Road, Jadavpur, Kolkata, West Bengal 700032, India

Introduction

Cancer is the second leading cause of death worldwide, causing around 9.6 million deaths every year and about 1/6th of the total deaths of the population throughout the globe. Studies also suggest that the economic impact of cancer is significant as most of these deaths have been reported from poor and middle-income countries where the living index is low and healthcare infrastructure is comparatively insufficient for diagnosis and potential treatments. Among them, cervical cancer is the fourth most common cancer worldwide, having around 570 thousands of reported cases every year and the second most common cancer in women causing around 311 thousands of deaths per year [14].

Pap smear test is currently the most reliable and renowned screening tool for the detection of pre-cancerous cells and cervical lesions in women based on the microscopic studies of the cells. However, the process is time-consuming since pathologists need a lot of time to classify each cell from a

slide of over 10,000 cells. Thus, an automated detection tool needs to be developed to classify pre-cancerous lesions for early detection and widespread screening.

With the advent of artificial intelligence and deep learning in the domain of medical sciences and healthcare [4], it is more becoming to lie on the results predicted by this decision-support system [5] to undermine the observer-bias issues. In this paper, we seek to develop an alternative approach that utilizes the deep learning-based feature extraction and optimization algorithm that gives excellent multi-class classification accuracy, performing robustly and outperforming several existing methods like [6, 8, 12, 15, 28, 29, 37].

In this research, we extracted deep features [3] from pre-trained convolution neural network (CNN) models and concatenated the features to form a large feature space. This is followed by the application of the principal component analysis (PCA) method for dimensionality reduction of the feature space, while keeping most of the important features intact, for which we retained 99% of the variance of the feature space. Some of the features extracted from the CNN classifiers might be non-informative or might even lead to a higher misclassification rate. To eliminate such redundant, misleading features, generally, evolutionary optimization algorithms are preferred by researchers, but applying such algorithms directly to CNN-extracted features leads to computational wastage due to the very high dimensionality of the feature space. PCA reduces the dimensionality of the feature space or number of data points by combining highly correlated variables to form a smaller number of new variables, known as "principal components". From PCA, we also get higher variations in the data, even in a lower dimension. This new feature space of dimensionality lower than the original space, when used as an input to a metaheuristic optimization algorithm, significantly reduces the computation cost since a lower population size is required for faster convergence. The number of iterations required for reaching the global optima is also greatly reduced (i.e., faster evolution).

Specifically, among different evolutionary optimization algorithms available, we used the grey wolf optimizer (GWO) [27] for optimal feature set selection. The GWO is embedded with a support vector machines (SVM) classifier [44] with radial basis function (RBF) kernel for fitness assignment and final classification. This method significantly decreased the training time for the classification task, while maintaining the competitive accuracy of predictions. The overall workflow of the proposed framework is shown in Fig. 1.

The contributions of this paper can be summarized as follows:

1. In the current paper, we propose a framework for the optimal selection of deep features extracted from convolutional neural network (CNN) classifiers.
2. The dimensionality of the feature set extracted from the CNNs is large and thus principal component analysis (PCA) is used to reduce the dimensionality while retaining the highly discriminating features. The resulting feature subset, when used as the input of GWO, reduces the computation and ensures faster convergence.
3. Optimal features are selected through the use of a nature-inspired evolutionary optimization algorithm, the grey wolf optimizer (GWO), for the first time in the cervical cytology domain, which filters out only the non-redundant features for making the final predictions.
4. The proposed framework has been evaluated on three publicly available datasets: the Herlev Pap Smear dataset [21], the SIPaKMeD Pap Smear dataset [31] and the

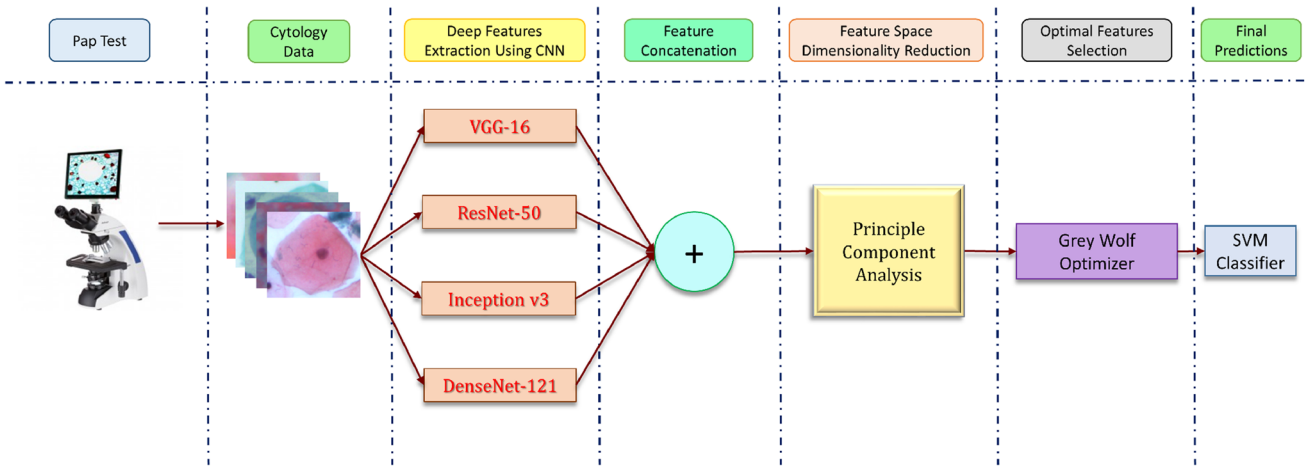

Fig. 1 Overall workflow of the proposed framework

Mendeley Liquid Based Cytology dataset [20], achieving classification accuracies of 98.32, 97.87 and 99.47%, respectively. The proposed method outperformed traditional CNN-based approaches and is comparable to state-of-the-art methods.

The rest of the paper is organized as follows: "Related Work" surveys some of the recent developments in the automated detection of cervical cancer; "Materials and Methods" describes the proposed methodology in detail; "Results and Discussion" evaluates the performance of the proposed framework on three publicly available benchmark datasets and "Conclusions and Future Work" concludes the paper.

Related Work

Previous studies show that feature extraction and selecting non-redundant features is an important part of the classification process and it affects the classification result significantly. Different methods have been explored over the years, like traditionally handcrafted feature extraction and feature selection [2], simulated annealing (SA) [34], convolutional neural networks [38], fuzzy-C means [35], to name a few. Some have given very good results in binary classification but not so much in multi-class classification, only a few have successfully given good results even for multi-class classification problem [6, 8, 24]. One of the major concerns are the availability of open-access datasets and the volume of images available in them for each class, which is a major setback for a lot of proposed methods in the literature.

Chankong et al. [8] used different classifiers for binary and multi-class classification with the best accuracy obtained using ANN. William et al. [36] used an enhanced Fuzzy C-means by extracting features from the cell images in Herlev dataset and got an accuracy of 98.88%. Byriel [7] used the ANFIS neuro-fuzzy classification technique which achieved an accuracy of 96% on the 2-class problem but performed much worse on the 7-class problem. Zhang et al. [42] used MLBC slides stained in H&E in two ways: once posing it as a 2-class dataset and again using it as a 7-class dataset. They used the Fuzzy Clustering-means approach and got an accuracy of 96–97% in the 2-class problem and 72–77% on the 7-class problem. Bora et al. [6] used Ensemble classifier on the Herlev dataset and got an accuracy of 96.5%. Marinakis et al. [23] in a way similar to [39, 42] of using the Herlev dataset as both a 2-class and a 7-class problem, used genetic algorithm combined with nearest neighbour classifier and got the best result with 1 nearest neighbour classifier giving an accuracy of 98.14% on the 2-class problem and 96.95% on the 7-class problem both in 10-fold cross-validation. Zhang et al. [43] used a deep convolutional neural network architecture thus removing the need for cell segmentation like [9, 19] on the Herlev dataset and achieved an accuracy of 98.3%.

However, the number of publicly available datasets related to the smear images of cervical cytology is quite less and each of them contains only nearly a thousand images or less. So it becomes quite difficult to design a classical deep learning or machine learning model with that few images to classify between these images with an improved accuracy than the preexisting methods. However, transfer learning can be used for this purpose that can quite significantly tackle this issue where we use a pre-trained model (for example ResNet-50 trained with ImageNet [11]), fine-tune the model, and use that for the classification purpose. Akter et al. [1] performed experimentations with different machine learning classifiers performed detailed comparative analysis on their performance. Data augmentation can be another solution where we can virtually increase the dataset size by slight movement or rotation or some other changes of the images. However, these methods cannot improve the results significantly as they cannot add more features or information to the algorithm to learn from. Therefore, as suggested by [30], we tried a new optimal feature selection approach for improving the classification accuracy and robustness of the task.

Materials and Methods

The experiment consisted of the following steps: (1) data acquisition (collecting image datasets of Pap smear test results from different sources), (2) data preprocessing (structuring the data incorrect formats and verifying the datasets), (3) feature extraction (extracting the important features from the datasets using different CNN models), (4a) combining the features from different CNNs (to increase the effectiveness of the features) (4b) feature reduction using principal component analysis (PCA) method (to discard the redundant features and to improve the classification time), (5) fitting these features to the classifier and (6) analysis of the results. The whole task was performed using a machine having NVIDIA Tesla K80 GPU with 12 GB of available RAM size.

Datasets Used

We use three publicly available cervical cytology datasets (Fig. 2) in this study for evaluating the proposed classification framework:

1. Herlev Pap Smear dataset by Jantzen et al. [21]
2. Mendeley Liquid Based Cytology dataset by Hussain et al. [20]
3. SIPaKMeD Pap Smear dataset by Plissiti et al. [31]

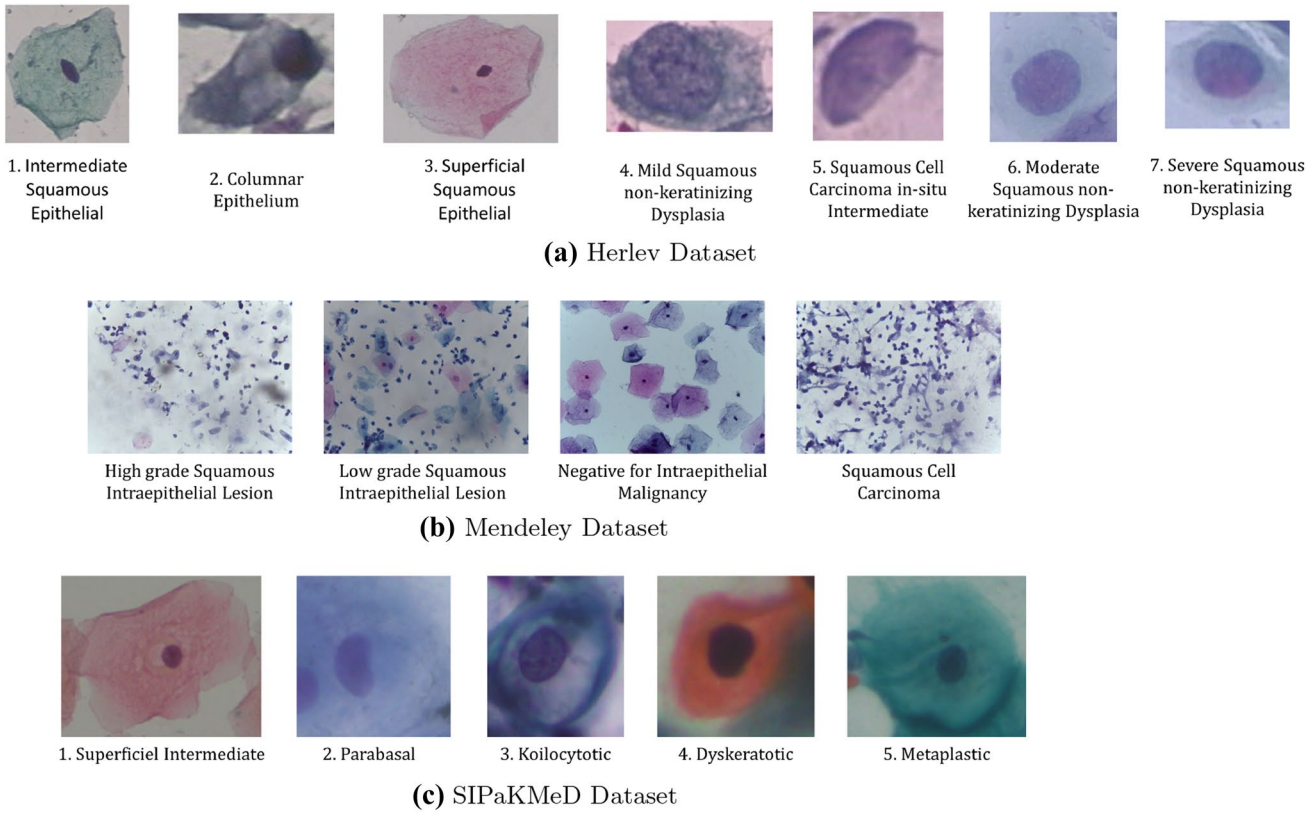

Fig. 2 Examples of images from each class in the three publicly available datasets

Table 1 Distribution of images in the three publicly available datasets

Dataset	Class	Category	Cell type	Number of images
Herlev Pap Smear (total: 917)	1	Normal	Intermediate squamous epithelial	70
	2	Normal	Columnar epithelial	98
	3	Normal	Superficial squamous epithelial	74
	4	Abnormal	Mild squamous non-keratinizing dysplasia	182
	5	Abnormal	Squamous cell carcinoma in-situ intermediate	150
	6	Abnormal	Moderate squamous non-keratinizing dysplasia	146
	7	Abnormal	Severe squamous non-keratinizing dysplasia	197
Mendeley LBC (total: 963)	1	Normal	Negative for intraepithelial malignancy	613
	2	Abnormal	Low grade squamous intraepithelial lesion (LSIL)	163
	3	Abnormal	High grade squamous intraepithelial lesion (HSIL)	113
	4	Abnormal	Squamous cell carcinoma (SCC)	74
SIPaKMeD Pap Smear (total: 4049)	1	Normal	Superficial-intermediate	831
	2	Normal	Parabasal	787
	3	Abnormal	Koilocytotic	825
	4	Abnormal	Dyskeratotic	813
	5	Benign	Metaplastic	793

These datasets are described in brief in the following subsections.

Herlev Pap Smear Dataset

The Herlev Pap Smear dataset is a publicly available benchmark dataset consisting of 917 single cell images distributed unevenly among 7 different classes. The distribution of images in each class are tabulated in Table 1.

Mendeley Liquid Based Cytology Dataset

The Mendeley LBC dataset [20] developed at Obstetrics and Gynecology department of Guwahati Medical College and Hospital, consists of 963 whole slide images of cervical cytology distributed unevenly in four different classes as shown in Table 1.

SIPaKMeD Pap Smear Dataset

The SIPaKMeD Pap Smear dataset by Plissiti et al. [31] consists of 4049 images of isolated cells (extracted from 966 whole slide images) categorized into five different classes based on their cytomorphological features. The distribution of images in the dataset is shown in Table 1.

Deep Features Extraction

Handcrafted or manual feature extraction using traditional Machine Learning techniques has limitations both in terms of the number of features and their correlations. Extracting features from a large dataset is a tedious task and can incorporate human biases, affecting the quality of the features that can eventually affect the classification task. Redundant features might be extracted which might lead to higher rates of misclassification. So, in this work, we extract deep features from CNN classifiers. Deep learning models use backpropagation to learn the important features themselves, and thus eliminates the tedious process of using handcrafted features. For the present study, we have used ResNet-50 [16], VGG-16 [32], Inception v3 [33] and DenseNet-121 [18] for extraction of features from the penultimate layer of the models.

While performing feature extraction from a CNN, we use the pre-trained model and fine-tune the CNN using our data, letting each image propagate through the layers in a forwarding direction, terminating at the pre-final layer, and taking out the output of this layer as the feature vector. We use pre-trained weights (Transfer Learning) in this study because the biomedical data is scarce and insufficient for deep learning models to work efficiently if trained from scratch. ImageNet [11] dataset consists of 14 billion images divided into 1000 classes. We use the models pretrained on this dataset and replace the final classification layer of size 1000 with a layer of size equals to the number of classes in our dataset. A model pretrained on such a large dataset already has learned important features from image data, and just needs fine-tuning for less number of epochs to train the final classification layer that we added.

VGG-16

The main characteristics of VGG nets [32] includes the use of 3×3 convolution layers which gave a noticeable improvement in network performances while making the network deep. 3×3 receptive filters were used throughout the entire net with strides of 1. Local response normalization (LRN) is not used in VGG Nets because memory consumption is more in such cases. The small-sized convolution filters give VGG Nets a chance to have a very large number of weight layers which in turn boosts performance. The input has a shape of $224 \times 224 \times 3$. In the present work, we fine-tune the VGG-16 model using our datasets, employing Stochastic Gradient Descent (SGD) optimizer and Rectified Linear Unit (ReLU) activation function.

ResNet-50

The ResNet-50 architecture [16] consists of residual skip connections embedded that make the training of the network easier. The gradient vanishing problem is addressed at the same time due to the embedding of the skip connections, which allows very deep networks to be accommodated for a controlled computation cost. $224 \times 224 \times 3$ sized inputs are used in the ResNet-50 model, with SGD optimizer and ReLU activation function.

Inception v3

The salient feature of the Inception v3 architecture [33], is the inception blocks that use parallel convolutions followed by channel concatenation. This leads to vivid features being extracted but with seemingly shallow networks. Parallel convolutions also allow the overfitting problem to be addressed while controlling the computational complexity. Inputs of shape $299 \times 299 \times 3$ are used with SGD optimizer and ReLU activation function for deep features extraction.

DenseNet-121

The DenseNet model [18] was proposed to address the vanishing gradient descent problem. The fundamental blocks in the DenseNet architecture are connected densely to each other leading to low computational requirement since the number of trainable parameters decreased heavily. The DenseNet architectures add small sets of feature maps owing to its narrow architecture. We used the DenseNet-121 variant with the SGD optimizer and ReLU activation function for deep features extraction.

Principle Component Analysis

Principal component analysis (PCA) is a linear dimensionality reduction method that transforms the higher dimensional data into a lower dimension by maximizing the variance of the lower dimension. PCA was first introduced in 1987 [44] however, further development and implementation of PCA in machine learning problems was done quite significantly in the later period. The covariance matrix of the feature vector is computed first and followed by the computation of eigenvectors of this matrix. The eigenvectors that have the largest eigenvalues contribute to the formation of new reduced dimensionality of the feature vector. Thus, instead of losing some of the important features of the data, we kept the most important of the features by preserving 99% of the variance. Before applying the PCA algorithm for feature dimension reduction, we need to perform data preprocessing that is required for the further steps. Depending upon the n-dimensional training set $x^{(1)}, x^{(2)}, x^{(3)}, \ldots x^{(n)}$, we need to perform mean normalization or feature scaling similar to the supervised learning algorithms. The mean of each feature is computed as in Eq. 1

$$\mu_i = \frac{1}{n}\sum_{j=1}^{n} x_i^{(j)}. \tag{1}$$

Now, we replace each of the x_i value with the $x_i - \mu_i$ value so that each of them has exactly zero mean value, however, if different features have different mean values, we can scale them so that they belong in a comparable range. In supervised learning, this scaling process of the ith element is defined by Eq. 2, where, s_i is the $|\max - \text{mean}|$ value or the static deviation of ith feature:

$$x_i^{(j)} = \frac{x_i^{(j)} - \mu_i}{s_i}. \tag{2}$$

For reducing the dimension of the feature from N to m (where $m < N$), and to define the surface in N-dimensional space onto which we project the data, we need to find the mean square error of the projected data on the m dimensional vector. The computational proof of the calculation of these m vectors: $u^{(1)}, u^{(2)}, \ldots, u^{(m)}$ and the projected points: $zu^{(1)}, z^{(2)}, \ldots, z^{(N)}$ on these vectors is complicated and beyond the scope of this paper. The covariance matrix is computed as in Eq. 3 where, the $x^{(j)}$ vector has $N \times 1$ dimension and $(x^{(j)})^{\mathrm{T}}$ has $1 \times N$ dimension, thus making the covariance matrix of the dimension of $N \times N$. Next, we calculate the eigenvalues and eigenvectors of the covariance matrix which represent the new magnitude of the feature vectors in the transformed vector space and their corresponding directions respectively. The eigenvalues quantify the variance of all the vectors as we are dealing with the covariance matrix. If an eigenvector has high valued eigenvectors, that means that it has high variance and contains various important information about the dataset. On the other hand, eigenvectors will small eigenvalues contain very small information about the dataset:

$$\text{covariance matrix} = \frac{1}{N}\sum_{j=1}^{N} x^{(j)} \times (x^{(j)})^{\mathrm{T}}. \tag{3}$$

Hence the pth complete principal component of a data vector $x^{(j)}$ in the transformed coordinates can be allocated a score $t^{(p)} = x^{(j)} \times w^{(p)}$ where $w^{(p)}$ is the pth eigenvector of the covariance matrix of $x^{(j)}$. Therefore the full PCA decomposition of the vector X can be represented as $T = X \times W$, where W is the eigenvector of the covariance matrix. Now, we need to select m-number of eigenvalues from these N eigenvectors by maximizing the variance of the preserved original data while reducing the total square reconstruction error. Next, we calculate the cumulative explained variance (CEV) which is the sum of variances (information) containing in the top m principal components. Then we set a threshold value above which, the eigenvalues will be considered as useful and the rest will be discarded as unimportant features. For our experiment we have set the threshold value to 99, meaning that we have kept 99% of the variance of the data retained in the reduced feature vector. As different CNN extracts features of different modalities, the number of selected features after PCA and GWO are different based on the feature distribution in those feature sets.

The pseudo-code of dimensionality reduction using PCA is shown in Algorithm 1.

Algorithm 1: Pseudo-code for Principal Component Analysis

define_function: PCA
Input:
Feature set X of dimension d
Compute Co-variance matrix Π

while (i≤ d) **do**

while (j ≤ d) **do**
$\mu_i \longleftarrow$ sample mean of feature i
$\mu_j \longleftarrow$ sample mean of feature j
$\sigma_{ij} = \frac{1}{n}\sum_{k=1}^{n}(x_i^k - \mu_i)(x_j^k - \mu_j)$
j=j+1
end while
$i = i + 1$
end while
decompose π into eigenvalues and eigenvectors
calculate cumulative explained variance (CEV)

if (CEV≥ threshold) **then**
Construct projection matrix W
end if
transform input X using W
obtain k-dimensional feature subspace X'
return X'

Grey Wolf Optimizer

Grey wolf optimization (GWO) [27] is a nature-inspired meta-heuristic optimization algorithm that mimics the leadership hierarchy of grey wolf (*Canis lupus*) and their hunting process for the optimization. Four types of GWO agents are deployed for simulation of the optimization algorithm named alpha, beta, delta, and omega. They mimic the three-step hunting methods of the grey wolf: finding the prey, encircling them, and finally attacking them for the sake of optimization.

The grey wolves follow the leadership of the alpha wolf, which is the topmost category of their strict social hierarchy. The alpha wolf is not necessarily the strongest and the fittest ones, but they can maintain the discipline of the whole pack. The major decisions are taken by the alpha wolves but often accompanied by the subordinates, the beta wolf. They are the next lower level of wolves in their social hierarchy and convey the decisions o the alpha to the lower levels of wolves. They are generally the fittest candidates for alpha if the alpha becomes old or weak and plays a major role in maintaining the pack as a whole. The next level of wolves is called delta and they play a very major role in decision-making and other important activities of the pack. The last and the least important category of the pack is named omega and they often play the role of scapegoat in society. Thus the complete pack is formed based on dominance hierarchy.

The mathematical model for the steps of optimization that mimics their hunting process is described below.

Social Hierarchy

Similar to the social hierarchy of grey wolf, the optimizer allocates the three fittest solutions as alpha, beta, and delta and the rest of the search agents are bound to arrange them and adjust accordingly as the parameters of the alpha, beta, and delta wolves. These three wolves are followed by the omega wolves.

Encircling the Prey (Optimal Solution)

To mathematically represent the encircling of prey, Eqs. 4, and 5 are used where t is the present iteration and $\boldsymbol{C}$ and $\boldsymbol{A}$ are the coefficient vectors, $\boldsymbol{x_p}$ indicates the vector position of the prey and $\boldsymbol{x}$ indicates the vector position of the grey wolf:

$$\boldsymbol{D} = |\boldsymbol{C} \times \boldsymbol{x_p}(t) - \boldsymbol{x}(t)| \tag{4}$$

$$\boldsymbol{x}(t+1) = \boldsymbol{x_p}(t) - \boldsymbol{A} \times \boldsymbol{D}. \tag{5}$$

The expression for $\boldsymbol{A}$ and $\boldsymbol{C}$ areas in Eqs. 6 and 7 respectively, where $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are the random valued vectors between 0 and 1 inclusive and the value of $\boldsymbol{a}$ decreases from 2 to 0 linearly with increase in iteration. The two random

variables $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ allow a grey wolf agent to reach any position between the points. The agents of the grey wolf algorithm can position themselves around the fittest solution by adjustment of the value of $\boldsymbol{C}$ and $\boldsymbol{A}$:

$$\boldsymbol{A} = 2 \times \boldsymbol{a} \times \boldsymbol{r_1} - \boldsymbol{a} \tag{6}$$

$$\boldsymbol{C} = 2 \times \boldsymbol{r_2}. \tag{7}$$

The same is applicable for n-dimensional optimization where the grey wolf agents move along the hyper-sphere or hyper-cubes around the fittest solution obtained.

Reaching the Optimal Solution

In real life, grey wolves hunt for their prey being led by the alpha who is accompanied by beta and delta wolves and the other wolves follow their instructions. To simulate this hunting principle, we allow some random valued agents and find their fitness and consider the three most accurate results as alpha, beta, and delta as in abstract search space we have no idea about the position of the agents and the prey. The rest of the agents including the omega wolves are bound to change their positions and orientations according to the three best wolf agents:

$$D_\alpha = |\boldsymbol{C_1} \times \boldsymbol{x}_\alpha - \boldsymbol{x}| \tag{8}$$

$$D_\beta = |\boldsymbol{C_2} \times \boldsymbol{x}_\beta - \boldsymbol{x}| \tag{9}$$

$$D_\delta = |\boldsymbol{C_3} \times \boldsymbol{x}_\delta - \boldsymbol{x}| \tag{10}$$

$$\boldsymbol{x_1} = \boldsymbol{x}_\alpha - \boldsymbol{A_1} \times \boldsymbol{D}_\alpha \tag{11}$$

$$\boldsymbol{x_2} = \boldsymbol{x}_\beta - \boldsymbol{A_2} \times \boldsymbol{D}_\beta \tag{12}$$

$$\boldsymbol{x_3} = \boldsymbol{x}_\delta - \boldsymbol{A_3} \times \boldsymbol{D}_\delta \tag{13}$$

$$\boldsymbol{x}(t+1) = \frac{\boldsymbol{x_1} + \boldsymbol{x_2} + \boldsymbol{x_3}}{3}. \tag{14}$$

The search agents update their position by these equations, however, the final position of the agents are not predefined, rather they are the random positions according to the position of the alpha, beta, and delta agents and within a certain circle which is determined by the position of the three best-fit solutions.

Exploiting the Prey

Grey wolves encircle their prey until the prey stops movement and this freezing the prey is known as exploiting. In the mathematical model, the value of $\boldsymbol{a}$ is decreased with the agents approaching the prey, and hence the value of $\boldsymbol{A}$ is modified further. The fluctuation of $\boldsymbol{A}$ is stopped as the value of $\boldsymbol{A}$ changes from $\boldsymbol{-2a}$ to $\boldsymbol{+2a}$. The value of $\boldsymbol{a}$ changes from 2 to 0 with the increase in iterations. The search agents can take any position between their current position and the position of the prey as the alpha, beta, and delta wolves approach the prey for hunting.

Exploring for the Prey

The grey wolf agents diverge in search of prey and they finally converge for attacking the prey. In mathematical modelling, this phenomenon is regulated by the value of $\boldsymbol{A}$; if the value of $\boldsymbol{A}$ is greater than 1 or less than -1, the grey wolf agents diverge from each other and find for some more suitable prey. However, if $\boldsymbol{A}$ has a value between -1 and $+1$, the agents converge towards the prey.

The overall pseudo-code of the GWO algorithm is shown in Algorithm 2 and the flowchart for the algorithm is shown in Fig. 3.

Algorithm 2: Pseudo-code for the Grey Wolf Optimizer for feature selection.

define_function: GWO
Input:
Number of Search Agents: n
Maximum number of iterations: MAX_ITER
Initialize the GWO population X_i $\forall i = 1, 2, 3, ..., n$
Initialize a, A, and C `// According to Equations 6,7`
Calculate the fitness of each search agent
X_α = Alpha wolf (Best search agent)
X_β = Beta wolf (Second best search agent)
X_δ = Delta wolf (Third best search agent)

while t < MAX_ITER **do**

for each search agent **do**
Update position of current search agent `// According to Equation 5`
end for
Update a, A, and C
Calculate the fitness of each search agent
Update $X_\alpha, X_\beta, X_\delta$ `// According to Equations 11, 12 & 13`
$t = t + 1$
end while
return X_α

Classification

After optimization, the final step is to fit the selected features to the classifier for the classification task. Due to a large number of features in some cases, we used incremental learning where a small batch of the dataset is selected for training the classifier and the loop over all the dataset and continue training until we reach convergence. This is fast and computationally efficient. We used an SVM classifier with the "RBF" kernel for the multi-class classification task.

Support Vector Machine

SVM [44] is a supervised learning model, which, in a set of training examples, properly labelled with different classes, add new examples to each class making a complete non-probabilistic binary classifier out of this SVM, and is associated with some typical learning algorithms which analyse the data, specifically used for regression and classification tasks. SVM model representation of the training samples in the feature plane is such that a separation between the

Algorithm 3: Pseudo-code for overall workflow.

Input:
Raw RGB images: I_{RGB}
f_i=deep features extracted from i^{th} CNN; $i = 1, 2, 3, 4$
$f_c = concat(f_1, f_2, \ldots, f_i); i = 1, 2, 3, 4$ `// concatenation of features`
$F_{PCA} = PCA(f_c)$ `// Using algorithm 1`
$F_{GWO} = GWO(F_{PCA})$ `// Using Algorithm 2`
train-test split$\rightarrow train_{F_{GWO}}, test_{F_{GWO}}$
$clf = classifier.fit(train_{F_{GWO}})$ `// Train SVM Classifier`
$predicted = clf.predict(test_{F_{GWO}})$ `// Make predictions on test set`
Compare predictions and labels and evaluate performance

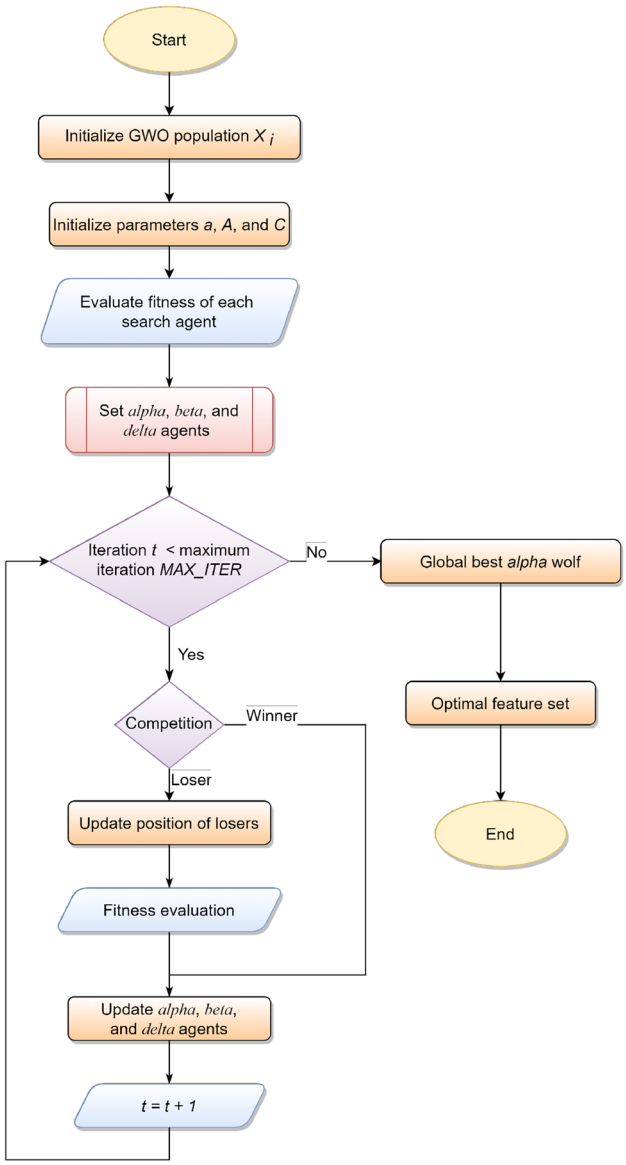

Fig. 3 Flowchart showing the workflow of the Grey Wolf Optimization algorithm used in the proposed framework

examples belonging to different classes becomes so prominent, that a curve can be fit in that space between two classes which maintain maximum distances from every point of each class and SVM fits that curve.

Results and Discussion

After extracting the features from the dataset using the CNN architectures said in "Materials and Methods", and the features were concatenated. We then use PCA (which retained 99% of the variance of the data) for the reduction in the dimensionality of the feature space and improvements in feature qualities respectively. Table 2 shows the statistics of reduction in feature dimensionality as well as the improvement of training time after this for the Herlev dataset. Then, we used the GWO algorithm and finally split the dataset and calculated the accuracy score for the training, validation, and testing sets. The overall workflow is shown in the form of pseudo-code in Algorithm 3. The results of our experiments are discussed in this section.

The metrics used for performance evaluation of the classification task for the multi-class problem is calculated based on Eqs. 15–18 which are derived from a confusion matrix M:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} M^{ii}}{\sum_{i=1}^{N}\sum_{j=1}^{N} M^{ij}} \tag{15}$$

$$\text{Precision}^i = \frac{M^{ii}}{\sum_{j=1}^{N} M^{ji}} \tag{16}$$

$$\text{Recall}^i = \frac{M^{ii}}{\sum_{j=1}^{N} M^{ij}} \tag{17}$$

$$\text{F1-score}^i = \frac{2}{\frac{1}{\text{Precision}^i} + \frac{1}{\text{Recall}^i}}. \tag{18}$$

To cross-validate the results of the classification task on different datasets and different features, we performed an AUC-ROC test on different datasets. The ROC (Receiver Operating Characteristics) curve (Fig. 4) is an important analyzing tool for validating the clinical findings of our experiment. The different line segments in the OVA (One Vs. All) ROC represent different classes stating that how good the features and the classifier performance are for classifying the different classes which can be broadly categorized in normal and infected cases. It represents the graphical analysis of the TPR (True Positive Rate) against the FPR (False Positive Rate) as the two operating characteristics criterion of the classifier based on the features selected. A false-positive result is a case when data of a healthy or uninfected class is predicted as an unhealthy or infected case by a classifier and it's a major drawback of the classification task. This is reciprocated by the points lying far above the diagonal line of the ROC curve suggesting that the TPR is significantly high as compared to FPR. Another important feature for analyzing the classification result is the AUC (Area Under Curve) of the ROC curve which was computed considering the 97% of the confidence interval. The analysis using the AUC-ROC curves for different datasets and different features are discussed further.

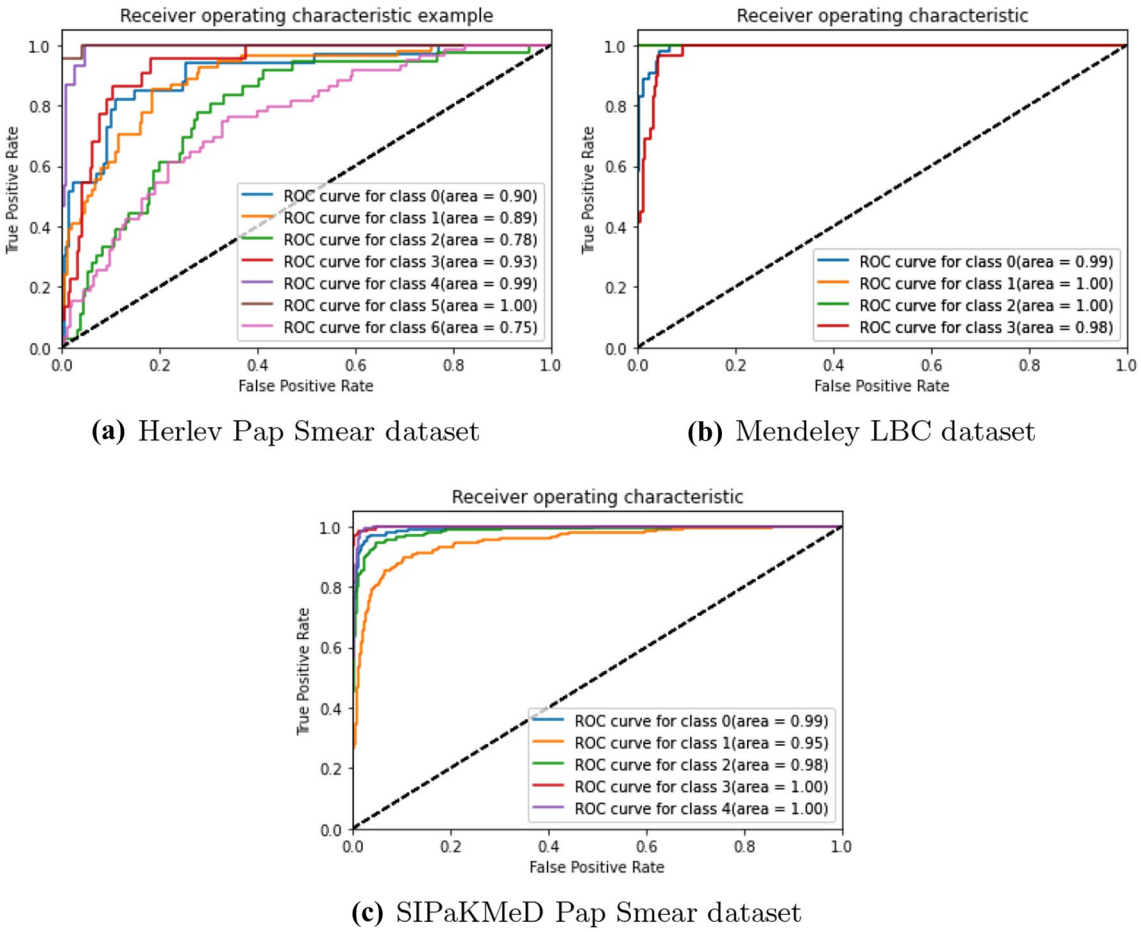

Fig. 4 ROC curves obtained by the proposed method for the three datasets: **a** Herlev Pap Smear dataset, **b** Mendeley LBC dataset and **c** SIPaKMeD Pap Smear dataset

Table 2 Reduction in feature dimension and improvements in training time after principal component analysis on the Herlev dataset

Model used for feature extraction	No. of features (before PCA)	No. of features (after PCA)	Reduction in feature dimension (%)	Improvement in average training time (%)
ResNet-50 [16]	100,353	383	99.62	88.425
VGG-16 [32]	25,088	364	98.55	85.215
DenseNet-121 [18]	50,177	330	99.34	81.449
Inception v3 [33]	131,073	325	99.75	84.228
ResNet-50 + VGG-16	125,441	456	99.63	80.221
DenseNet-121 + Inception v3	181,250	687	99.62	82.694
ResNet-50 + VGG-16 + DenseNet-121 + Inception v3	306,691	796	99.74	85.737

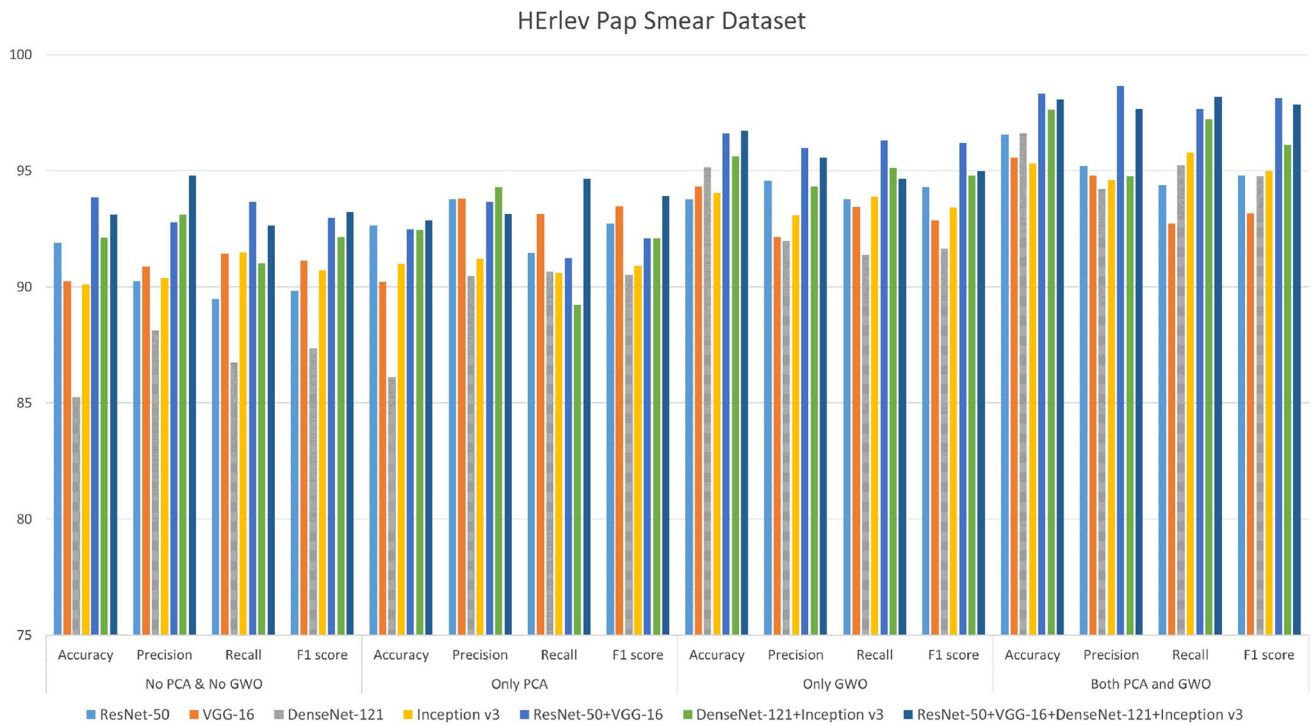

Fig. 5 Results on the Herlev Pap Smear dataset

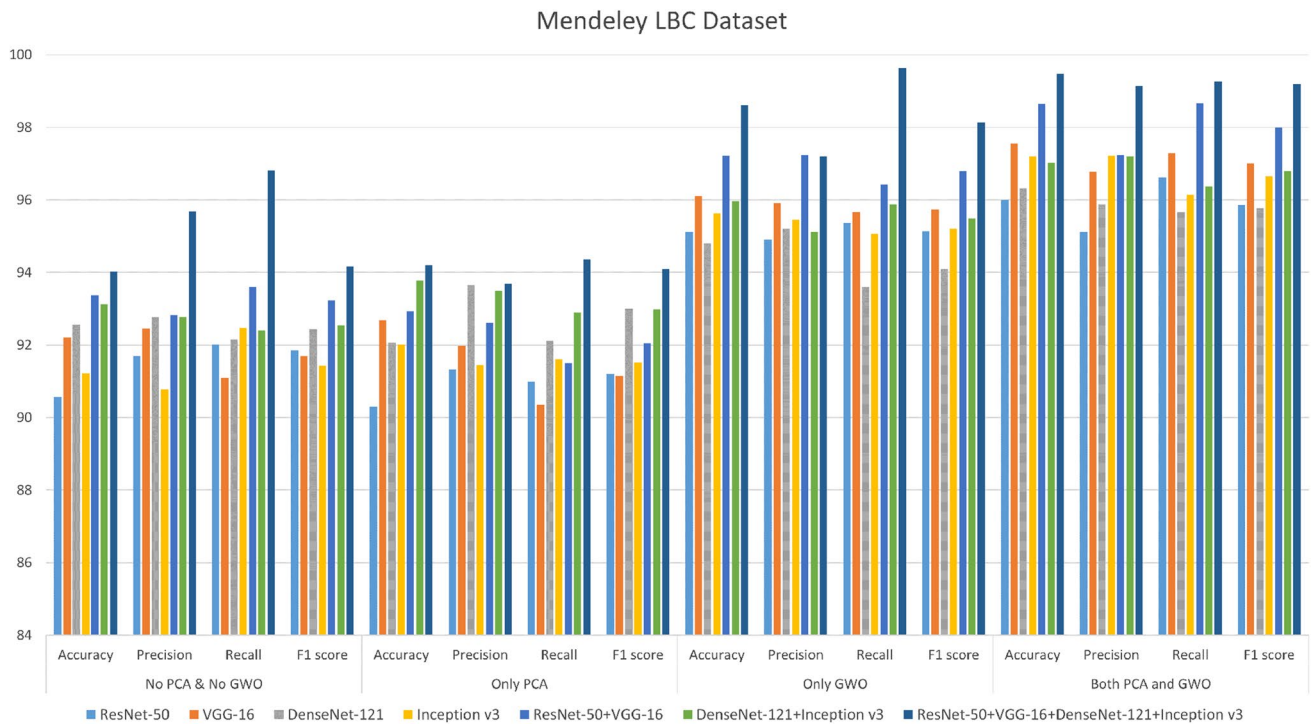

Fig. 6 Results on the Mendeley LBC dataset

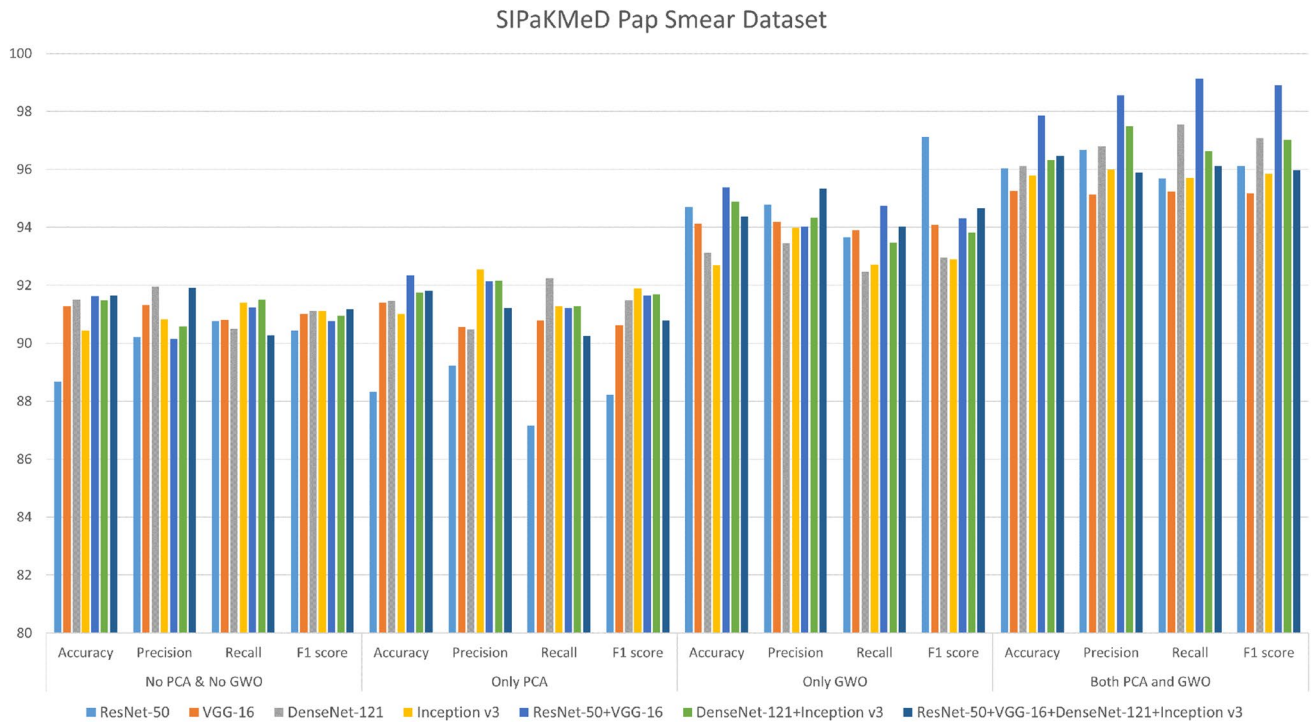

Fig. 7 Results on the SIPaKMeD Pap Smear dataset

Results on Herlev Pap Smear Dataset

The results obtained on different experiments on the Herlev Pap Smear dataset are shown in Fig. 5. The best classification results observed this dataset was achieved by merging the feature extracted from ResNet-50 and VGG-16 models, which gave the performance metrics as follows: accuracy = 98.32%, precision = 98.66%, recall = 97.65% and F1-score = 98.12%.

Results on the Mendeley LBC Dataset

The results obtained on different experiments on the Mendeley LBC dataset are shown in Fig. 6. The best results on this dataset are obtained by merging features extracted from VGG-16, ResNet-50, Inception v3 and DenseNet-121: accuracy = 99.47%, precision = 99.14%, recall = 99.27% and F1-score = 99.20%.

Results on SIPaKMeD Pap Smear Dataset

The results obtained on different experiments on the SIPaKMeD Pap Smear dataset are shown in Fig. 7. The best results on the dataset are obtained by merging features extracted from VGG-16 and ResNet-50: accuracy = 97.87%, precision = 98.56%, recall = 99.12% and F1-score = 98.89% (Tables 3, 4).

Comparison with Existing Literature

Several models have been proposed in the literature for cervical cell classification as discussed in "Related Work". Our proposed work and the results achieved are therefore compared with some of these models that used the same datasets to assess the reliability of the proposed framework and the results are tabulated in Table 5. No papers as of yet have been published that use the Mendeley LBC dataset, and thus we are unable to compare our method in that dataset.

McNemar's Statistical Test

The McNemar's statistical test has been performed in the present work, for the statistical analysis of the proposed classification framework. For this, the proposed model has been compared to the CNN models from which the features were extracted and used for the final classification. The results are shown in Table 6. To reject the null hypothesis that two models are similar, the p-value from the McNemar's test should remain below 5% (i.e., 0.05), and from the table, it can be seen that for every comparison case, the p-value < 0.05. Thus, the null hypothesis can be rejected and it can be concluded that the proposed model is dissimilar to any of the feature extractor models and performs superior to them. Thus statistical analysis of the proposed model justifies the reliability of the approach devised in this research.

Table 3 Accuracies and Losses on training, validation and testing sets using both PCA and GWO on the three datasets dataset (all the accuracy measurements are in % and measured after 30 epochs)

Dataset	Feature extractor model	Training accuracy	Training loss	Validation accuracy	Validation loss	Testing accuracy	Testing loss
Herlev Pap Smear	ResNet-50 [16]	97.77	0.026	96.33	0.032	96.55	0.028
	VGG-16 [32]	96.36	0.031	94.21	0.109	95.56	0.081
	DenseNet-121 [18]	97.89	0.02	97.01	0.024	96.61	0.024
	Inception v3 [33]	96.33	0.032	95.17	0.098	95.32	0.094
	ResNet-50 + VGG-16	**98.77**	**0.011**	**98.00**	**0.019**	**98.32**	**0.016**
	DenseNet-121 + Inception v3	97.91	0.026	96.01	0.031	97.62	0.027
	ResNet-50 + VGG-16 + DenseNet-121 + Inception v3	98.3	0.018	97.95	0.021	98.06	0.019
Mendeley LBC	ResNet-50 [16]	96.88	0.066	96.11	0.071	96	0.079
	VGG-16 [32]	97.91	0.051	96.39	0.068	97.56	0.059
	DenseNet-121 [18]	97.16	0.061	96.15	0.07	96.32	0.071
	Inception v3 [33]	97.5	0.058	97.05	0.06	97.2	0.061
	ResNet-50 + VGG-16	99.04	0.039	97.96	0.054	98.64	0.054
	DenseNet-121 + Inception v3	98.06	0.044	96.49	0.067	97.02	0.064
	ResNet-50 + VGG-16 + DenseNet-121 + Inception v3	**99.58**	**0.03**	**98.88**	**0.043**	**99.47**	**0.04**
SIPaKMeD Pap Smear	ResNet-50 [16]	96.85	0.028	96.77	0.049	96.03	0.048
	VGG-16 [32]	96.71	0.03	94.02	0.071	95.26	0.059
	DenseNet-121 [18]	96.31	0.035	96.04	0.058	96.12	0.046
	Inception v3 [33]	96.02	0.039	95.91	0.06	95.78	0.055
	ResNet-50 + VGG-16	**98.48**	**0.014**	**97.55**	**0.041**	**97.87**	**0.034**
	DenseNet-121 + Inception v3	97.32	0.02	95.39	0.066	96.33	0.044
	ResNet-50 + VGG-16 + DenseNet-121 + Inception v3	96.92	0.025	96.66	0.051	96.46	0.042

The bold values in the table indicates the best performance, that is, the performance obtained by the proposed method

Table 4 Comparison (ACC, in %) with standard optimization algorithms: *PSO* particle swarm optimization [22]; *MVO* mean variance optimization [13]; *GWO* grey wolf optimizer [27]; *MFO* moth flame optimization [25]; *WOA* whale optimization algorithm [26]; *FFA* firefly algorithm [40]; *BAT* bat optimization algorithm [41]; *GA* genetic algorithm [10, 17]

Optimization algorithms	Mendeley LBC dataset		Herlev Pap-smear dataset		SIPaKMeD 5-class dataset	
	ACC	# of features	ACC	# of features	ACC	# of features
PSO	95.90	920	92.58	992	90.14	1014
MVO	96.91	720	94.26	764	90.48	843
GWO	92.14	810	92.40	807	89.98	791
MFO	94.20	803	93.19	851	90.58	832
WOA	95.08	843	92.36	847	90.58	802
FFA	94.42	715	92.46	820	89.56	792
BAT	95.64	857	94.58	762	90.21	749
GA	98.23	724	95.26	784	95.43	796
PCA + GWO	**99.47**	**762**	**98.32**	**796**	**97.87**	**736**

The bold values in the table indicates the best performance, that is, the performance obtained by the proposed method

Table 5 Comparison of the proposed method with existing literature

Dataset	Method	Results
Herlev Pap Smear	GençTav et al. [15]	Precision: 88%0.15 Recall: 93%0.15
	Bora et al. [6]	Accuracy: 96.51%
	Win et al. [37]	Accuracy: 90.84%
	Chankong et al. [8]	Accuracy: 93.78%
	Proposed method	**Accuracy: 98.32%** **Precision: 98.66%** **Recall: 97.65%** **F1-score: 98.12%**
SIPaKMeD Pap Smear	Win et al. [37]	Accuracy: 94.09%
	Plissiti et al. [31]	1. Deep convolutional + SVM: 93.35%0.62 2. Deep fully connected + SVM: 94.44%1.21 3. CNN: 95.35%0.42
	Proposed method	**Accuracy: 97.87%** **Precision: 98.56%** **Recall: 99.12%** **F1-score: 98.89%**

The bold values in the table indicates the best performance, that is, the performance obtained by the proposed method

Table 6 Results obtained from McNemar's statistical test. For all three datasets, the proposed framework is compared to the CNN models whose features have been used

McNemar's test	*p*-value		
Performed with	Herlev Pap Smear	Mendeley LBC	SIPaKMeD Pap Smear
ResNet-50	0.0046	0.0012	0.0005
VGG-16	0.0001	0.0211	0.0007
DenseNet-121	0.0103	0.0089	0.0315
Inception v3	0.0007	0.0061	0.0100

The *p*-value is less than 0.05 for every case and thus, the null hypothesis is rejected

Conclusions and Future Work

The need for automation in the cervical cancer detection domain arises due to the high mortality rate throughout the close. Motivated by this cause, we developed a fully automated detection framework that optimizes deep features for classification. The two-level enhancement boosted the classification performance while simultaneously reducing the training time significantly. This research also explored the hybridization of multiple CNN-based deep features to extract more discriminating information from the dataset.

An alternative way of feature selection is exalted in this research that uses principal component analysis (PCA) and Grey Wolf Optimization (GWO). The two-level feature reduction approach introduced in this paper leverages the advantages of both methods resulting in optimal feature set selection. The proposed method achieves better results juxtaposed to end-to-end classification with CNN models, while simultaneously reducing the computation cost. Very high classification accuracy of 99.47, 98.32, and 97.87% on the three publicly available benchmark datasets, namely Mendeley LBC, Herlev Pap Smear and SIPaKMeD Pap Smear datasets, respectively, tantamount to state-of-the-art methods.

However, there is scope for further improvement by utilizing different classification models and using hybrid metaheuristic feature selection algorithms. This paper craved a path for further research in this field as well as multi-domain adaptation. The proposed pipeline can be used as a test-bed for several classification problems, not only in biomedical applications but in other computer vision problems as well. The feature selection can be further addressed by developing an end-to-end multi-objective hybrid optimization algorithm, that selects optimal feature set, where the objective function aims to increase the classification performance by selecting the least number of features, thereby reducing the computational cost simultaneously.

Acknowledgements The work is supported by SERB (DST), Govt. of India (Ref. no. EEQ/2018/000963).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Akter L, Islam MM, Al-Rakhami MS, Haque MR, et al. Prediction of cervical cancer from behavior risk using machine learning techniques. SN Comput Sci. 2021;2(3):1–10.
2. AlMubarak HA, Stanley J, Guo P, Long R, Antani S, Thoma G, Zuna R, Frazier S, Stoecker W. A hybrid deep learning and handcrafted feature approach for cervical cancer digital histology image classification. Int J Healthc Inf Syst Inform. 2019;14(2):66–87.
3. Azaza A, Abdellaoui M, Douik A. Off-the-shelf deep features for saliency detection. SN Comput Sci. 2021;2(2):1–10.
4. Basak H, Kundu R. Comparative study of maturation profiles of neural cells in different species with the help of computer vision and deep learning. In: International symposium on signal processing and intelligent recognition systems. Springer; 2020. p. 352–66.
5. Basak H, Kundu R, Agarwal A, Giri S. Single image super-resolution using residual channel attention network. In: 2020 IEEE 15th international conference on industrial and information systems (ICIIS). IEEE; (2020). p. 219–24.
6. Bora K, Chowdhury M, Mahanta LB, Kundu MK, Das AK. Automated classification of pap smear images to detect cervical dysplasia. Comput Methods Programs Biomed. 2017;138:31–47.
7. Byriel J. Neuro-fuzzy classification of cells in cervical smears. Master's Thesis, Technical University of Denmark: Oersted-DTU, Automation. 1999.
8. Chankong T, Theera-Umpon N, Auephanwiriyakul S. Automatic cervical cell segmentation and classification in Pap smears. Comput Methods Programs Biomed. 2014;113(2):539–56.
9. Chattopadhyay S, Basak H. Multi-scale attention U-Net (MsAU-NeT): a modified U-Net architecture for scene segmentation. 2020. arXiv:200906911.
10. De Jong KA. Analysis of the behavior of a class of genetic adaptive systems. Technical report. 1975.
11. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE; 2009. p. 248–55.
12. Dey S, Das S, Ghosh S, Mitra S, Chakrabarty S, Das N. SynCGAN: using learnable class specific priors to generate synthetic data for improving classifier performance on cytological images. In: Communications in computer and information science. Springer Singapore; 2020. p. 32–42. https://doi.org/10.1007/978-981-15-8697-2_3.
13. Erlich I, Venayagamoorthy GK, Worawat N. A mean-variance optimization algorithm. In: IEEE congress on evolutionary computation. IEEE; 2010. p. 1–6.
14. Ferlay J, Colombet M, Soerjomataram I, Mathers C, Parkin D, Piñeros M, Znaor A, Bray F. Estimating the global cancer incidence and mortality in 2018: GLOBOCAN sources and methods. Int J Cancer. 2019;144(8):1941–53.
15. GençTav A, Aksoy S, ÖNder S. Unsupervised segmentation and classification of cervical cell images. Pattern Recognit. 2012;45(12):4151–68.
16. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770–8.
17. Holland JH, et al. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT Press; 1992.
18. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 4700–8.
19. Huang J, Wang T, Zheng D, He Y. Nucleus segmentation of cervical cytology images based on multi-scale fuzzy clustering algorithm. Bioengineered. 2020;11(1):484–501.
20. Hussain E, Mahanta LB, Borah H, Das CR. Liquid based-cytology Pap smear dataset for automated multi-class diagnosis of pre-cancerous and cervical cancer lesions. Data Brief. 2020;30:105589.
21. Jantzen J, Norup J, Dounias G, Bjerregaard B. Pap-smear benchmark data for pattern classification. In: Nature inspired smart information systems (NiSIS 2005); 2005. p 1–9.
22. Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol. 4. IEEE; 1995. p. 1942–8.
23. Marinakis Y, Dounias G, Jantzen J. Pap smear diagnosis using a hybrid intelligent scheme focusing on genetic algorithm based feature selection and nearest neighbor classification. Comput Biol Med. 2009;39(1):69–78.
24. Martínez-Más J, Bueno-Crespo A, Martínez-España R, Remezal-Solano M, Ortiz-González A, Ortiz-Reina S, Martínez-Cendán JP. Classifying Papanicolaou cervical smears through a cell merger approach by deep learning technique. Expert Syst Appl. 2020;160:113707.
25. Mirjalili S. Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl Based Syst. 2015;89:228–49.
26. Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw. 2016;95:51–67.
27. Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Softw. 2014;69:46–61.
28. Mitra S, Dey S, Das N, Chakrabarty S, Nasipuri M, Naskar MK. Identification of malignancy from cytological images based on superpixel and convolutional neural networks. In: Studies in computational intelligence. Springer Singapore; 2019. p. 103–22. https://doi.org/10.1007/978-981-13-7334-3_8.
29. Mitra S, Das N, Dey S, Chakrabarty S, Nasipuri M, Naskar MK. Cytology image analysis techniques towards automation: systematically revisited. 2020. arXiv:2003.07529.
30. Niedzielewski K, Marchwiany ME, Piliszek R, Michalewicz M, Rudnicki W. Multidimensional feature selection and high performance parallex. SN Comput Sci. 2020;1(1):1–7.
31. Plissiti ME, Dimitrakopoulos P, Sfikas G, Nikou C, Krikoni O, Charchanti A. SIPAKMED: a new dataset for feature and image based classification of normal and pathological cervical cells in Pap smear images. In: 2018 25th IEEE international conference on image processing (ICIP). IEEE; 2018. p. 3144–8.
32. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014. arXiv:1409.1556.
33. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna ZB. Rethinking the Inception Architecture for Computer Vision. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016). https://doi.org/10.1109/cvpr.2016.308.
34. Wang XY, Garibaldi JM. Simulated annealing fuzzy clustering in cancer diagnosis. Informatica. 2005;29:61–70.
35. William W, Ware A, Basaza-Ejiri AH, Obungoloch J. Cervical cancer classification from Pap-smears using an enhanced fuzzy C-means algorithm. Inform Med Unlocked. 2019;14:23–33.

36. William W, Ware A, Basaza-Ejiri AH, Obungoloch J. A pap-smear analysis tool (PAT) for detection of cervical cancer from pap-smear images. Biomed Eng Online. 2019;18(1):16.
37. Win KP, Kitjaidure Y, Hamamoto K, Myo Aung T. Computer-assisted screening for cervical cancer using digital image processing of Pap smear images. Appl Sci. 2020;10(5):1800.
38. Wu M, Yan C, Liu H, Liu Q, Yin Y. Automatic classification of cervical cancer from cytological images by using convolutional neural network. Biosci Rep. 2018;38(6). https://doi.org/10.1042/BSR20181769.
39. Xue D, Zhou X, Li C, Yao Y, Rahaman MM, Zhang J, Chen H, Zhang J, Qi S, Sun H. An application of transfer learning and ensemble learning techniques for cervical histopathology image classification. IEEE Access. 2020;8:104603–18.
40. Yang XS. Firefly algorithms for multimodal optimization. In: International symposium on stochastic algorithms. Springer; 2009. p. 169–78.
41. Yang XS, Gandomi AH. Bat algorithm: a novel approach for global engineering optimization. Eng Comput. 2012.
42. Zhang L, Kong H, Ting Chin C, Liu S, Fan X, Wang T, Chen S. Automation-assisted cervical cancer screening in manual liquid-based cytology with hematoxylin and eosin staining. Cytom Part A. 2014;85(3):214–30.
43. Zhang L, Lu L, Nogues I, Summers RM, Liu S, Yao J. DeepPap: deep convolutional networks for cervical cell classification. IEEE J Biomed Health Inform. 2017;21(6):1633–43.
44. Zhang Y. Support vector machine classification algorithm and its application. In: International conference on information computing and applications. Springer; 2012. p. 179–86.