Article

Exploration and Enhancement of Classifiers in the Detection of Lung Cancer from Histopathological Images

Karthikeyan Shanmugam and Harikumar Rajaguru *

Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam 638401, India; ms.karthi.1388@gmail.com
* Correspondence: harikumarrajaguru@gmail.com

Abstract: Lung cancer is a prevalent malignancy that impacts individuals of all genders and is often diagnosed late due to delayed symptoms. To catch it early, researchers are developing algorithms to study lung cancer images. The primary objective of this work is to propose a novel approach for the detection of lung cancer using histopathological images. In this work, the histopathological images underwent preprocessing, followed by segmentation using a modified approach of KFCM-based segmentation and the segmented image intensity values were dimensionally reduced using Particle Swarm Optimization (PSO) and Grey Wolf Optimization (GWO). Algorithms such as KL Divergence and Invasive Weed Optimization (IWO) are used for feature selection. Seven different classifiers such as SVM, KNN, Random Forest, Decision Tree, Softmax Discriminant, Multilayer Perceptron, and BLDC were used to analyze and classify the images as benign or malignant. Results were compared using standard metrics, and kappa analysis assessed classifier agreement. The Decision Tree Classifier with GWO feature extraction achieved good accuracy of 85.01% without feature selection and hyperparameter tuning approaches. Furthermore, we present a methodology to enhance the accuracy of the classifiers by employing hyperparameter tuning algorithms based on Adam and RAdam. By combining features from GWO and IWO, and using the RAdam algorithm, the Decision Tree classifier achieves the commendable accuracy of 91.57%.

Keywords: histopathology; benign; adenocarcinoma; PSO; GWO; KL divergence; IWO; multilayer perceptron; bayesian linear discriminant analysis classifier

1. Introduction

Cancer is increasingly common, and doctors use blood tests, biopsies, and image analysis for its diagnosis. It originates from damaged cells and varies among individuals. Understanding its source helps us comprehend its condition [1]. Lung cancer, often tied to smoking or harmful exposures, is a prevalent cancer type causing rising death tolls globally [2]. It affects both genders and has a low survival rate. Early detection is crucial for better outcomes. The five-year survival rate is approximately 34% for surgically removable early-stage cancer, compared to less than 10% for inoperable cases. Lung cancer treatment depends on histological characteristics, categorized as small cell (SCLC) and non-small cell (NSCLC) types, of which 80% to 85% are NSCLC and the rest are SCLC [3]. NSCLC has subtypes such as benign, adenocarcinoma (ACA), and squamous cell carcinoma (SCC). SCC displays characteristics such as the presence of clusters of polyhedral cells, keratinization, and the formulation of keratin pearls. Once the tissue type is identified, suitable treatments can be selected: either surgery, chemotherapy, radiation, targeted therapy, or immunotherapy.

Early detection and treatment of cancer are vital for better patient outcomes. Traditional diagnostic methods involve clinical assessments, lab tests, imaging, and a procedure called biopsy [4], which is considered the gold standard. During biopsy, tissue samples are taken and examined under a microscope using techniques such as hematoxylin and

eosin staining. This histopathological analysis helps identify abnormal tissue growth and cell characteristics. Accurate identification and classification of individual cell nuclei are of utmost importance when evaluating tissue samples for cancer diagnosis. Pathologists inspect these samples at different magnifications, looking for signs of malignancy such as irregular cell shape, dark nuclei, and increased mitotic figures [5], and count to generate reliable results [6]. Manual histopathological examination is a time-consuming process due to the frequent presence of numerous nuclei from diverse categories clustered together in histopathological images, which can result in disagreements among pathologists [7], prompting the development of automated systems. Researchers have utilized image processing, pattern recognition, and machine learning/deep learning techniques to create computer-aided diagnostic (CAD) systems. These systems aim to detect and classify carcinomas quickly and reliably [8]. Machine learning and deep learning algorithms improve CAD performance as they learn from more data. These approaches use either microscopic images or whole slide images (WSIs) and extract features to aid in diagnosis. The challenge is to create a novel, versatile, and fully automated CAD system that can handle both microscopic images and WSIs, regardless of any imaging artifacts. Automated analysis of microscopic images is vital for evaluating digitized specimens, reducing inter-observer variations, and improving objectivity and reproducibility, as emphasized by Foran et al. [9]. This advancement can enable comparative studies of diseases and potentially aid in diagnostic decision-making.

Different imaging techniques, such as ultrasounds, MRIs, CT scans, X-rays, and needle biopsies, are used to diagnose lung cancer. X-ray imaging, considered a fundamental technique for lung examination, possesses restricted resolution and the potential to overlook specific areas of interest [10]. CT scans are commonly used to detect early stages of lung cancer and locate tumors before surgery, but they expose patients to harmful radiation with repeated scans. MRI demonstrates notable sensitivity and specificity, valuable for identifying bone metastases, although it is not advisable for diagnosing lung cancer. Ultrasound, a non-invasive method, proves adept at identifying postoperative lung issues and surpasses X-rays in effectiveness [11]. While image examination aids in diagnosis, staging, treatment evaluation, and prognosis assessment, histopathological examination remains the most reliable method to determine tumor characteristics and clinical stages. Histopathological images offer an intricate view of cellular and tissue-level transformations linked in differentiating between various conditions and cancer types, empowering pathologists to deliver precise and reliable diagnoses. Moreover, they are invaluable for pinpointing distinct biomarkers linked to various cancer types and grades, facilitating tumor classification and subtyping. Histopathological images form a dependable diagnostic framework known for its consistency and reliability in cancer diagnosis [12]. By harnessing extensive datasets of annotated histopathological images, it becomes feasible to create highly dependable algorithms for automated cancer diagnosis. These algorithms effectively streamline the diagnostic process, reducing the necessity for extensive manual examination [13].

The objective of this study is to create a classification framework that can analyze histopathological images data related to lung cancer. The goal is to accurately classify individuals as either having cancer or not, using machine learning techniques and meta-heuristic algorithms for tasks such as feature extraction, feature selection, and classification. The following subsection analyzes various methods for cancer detection and classification using image processing and classification techniques.

Review of Previous Work

In recent times, the research community has shown significant interest in diagnosing Lung Cancer through histopathological images. Numerous methodologies have been explored, utilizing a range of machine learning and deep learning techniques, across diverse datasets to detect instances of lung cancer.

Various strategies have been proposed to identify irregularities in lung-related images, encompassing chest radiographs, CT scans, ultrasound images, histopathological images,

and microarray data. Ozekes and Camurcu [14] utilized template matching, while Schilham et al. devised a computer-aided detection (CAD) system that encompasses preprocessing, the identification of candidate nodules, feature extraction, and cancer classification [15]. Wang et al. [16] executed the classification of pathology images concerning lung cancer using a convolutional neural network (CNN) methodology, incorporating cell segmentation. The final layer of the CNN model integrated the Softmax activation function to enhance the classification process. Through the application of the region of interest (ROI) technique as a preliminary step, they focused on cell areas containing relevant tumors. The achieved classification accuracy for the three-class image dataset reached 90.1%. Dehmeshki et al. [17] employed a genetic algorithm based on shapes for template matching, while Suarez-Cuenca et al. used an iris filter for CT image discrimination [18]. Murphy et al. used a K-nearest neighbor (KNN) classifier for nodule detection [19], and Giger et al. used geometric features in their CAD system for CT images.

Wei et al. [20] undertook the categorization of histopathological images depicting six classes of lung cancer utilizing CNNs. They specifically employed ResNet models for their investigation. The ResNet models were integrated with pre-trained approaches from ImageNet and COCO image databases. Prior to the model training phase, the input data underwent preprocessing, which included the application of augmentation techniques. The study's achievement in terms of classification F-score reached a notable 90.4%. Mohammed Al-Jabber et al. [21] employed histopathological images from the LC25000 dataset, employing both ANN and the GoogLeNet and VGG-19 models. This combination yielded an impressive accuracy of 99.64%. Teramoto et al. [22] effectively distinguished histopathological images spanning three types of lung cancer through the application of a deep learning model. They implemented an augmentation approach that involved rotating, flipping, and applying filters to each image. Following this, they employed their developed deep CNN model to carry out the classification process. The outcomes of their classification efforts yielded an accuracy of around 70%. Shapcott et al. [23] conducted their model training by initially subjecting the input data to a preprocessing stage, integrating the augmentation technique. They employed a deep learning methodology for classifying histopathological images related to colon cancer. The dataset encompassed four distinct classes. To facilitate cell identification, a cell patches algorithm was employed on each image. The images were segmented into specific dimensions through segmentation procedures. The classification process was then conducted using the CNN model based on the defined cell patches. The obtained correlation accuracy rates ranged between 90% and 96.9%.

Barker et al. an automated system to classify brain tumors using digital pathology images [24]. Ojansivu et al. explored an automated method for categorizing breast cancer from tissue samples [25]. Ficsor et al. proposed an automated classification method for colon inflammation using digital microscopy images of histological sections [26]. The authors of a study, Mouelhi et al. [27], used various techniques such as Haralick's textures, histogram of oriented gradients (HOG), and color-based statistical moments (CCSM) to extract features from biopsy images and classify cancerous cells. The features included energy, correlation, homogeneity, contrast, GLCM texture features, as well as RGB, gray level, and HSV color components. Huang and Lai [28] focused on histology image analysis, employing texture features and KNN, SVM for image classification and segmentation. Their approach achieved a classification accuracy of 90.07% and 92.8%. Gessert et al. [29] executed the classification procedure employing CNN models based on transfer learning, leveraging microscopic images of colon cancer. Their study employed a dataset that comprised both benign and malignant images. They trained the dataset using various models including Inception, VGG, and DenseNet. Among these, the DenseNet model yielded the most promising classification outcome, achieving a classification accuracy of 91.2%.

Sinha and Ramkrishan [30] studied small biopsy images, analyzing cell characteristics such as shape, size, color, and other properties. Four classification methods were compared: Bayesian, KNN, neural networks, and SVM. The last two methods achieved the highest accuracy rates of 94.1%, while the first two had lower rates of 82.3% and 70.6%.

Kasmin et al. [31] examined microscopic biopsy images, considering characteristics such as cell/nuclei size, cell boundary length, minimum polygon area enclosing a cell, major axis length of an ellipse fitted to a cell, filled cell area, and average cytoplasmic intensity. They used neural networks and achieved classification accuracies of 86% and 92%. Chia-Hung Chen et al. [32] used a convolutional neural network to diagnose endobronchial ultrasound images, achieving an improved accuracy of 85.4% compared to traditional methods. Azka Khoirunnia et al. [33] developed a lung cancer detection system using a combination of CNN and RNN with Microarray data. In their research, CNN achieved 83% accuracy, RNN reached 71%, and the fusion of CNN and RNN (CRNN) attained the highest accuracy at 91%. Shahid Mehmood et al. [34] focused on classifying histopathological images of lung and colon cancers. By using AlexNet along with a technique called Class-Selective Contrast Enhancement, they achieved an impressive accuracy of 98.4%.

This paper is structured as follows: Section 2 focuses on the methodology employed for detecting lung cancer. Section 3 explores the feature extraction techniques including Particle Swarm Optimization and Grey Wolf Optimization whereas Section 4 explores the feature selection techniques, such as KL Divergence, and Invasive Weed Optimization. Section 5 explains the different classifiers used and hyper parameter updating method and its implementation. Section 6 presents the cumulative results, and Section 7 concludes the paper.

The following section deals with the methodology employed for identifying lung cancer through histopathological images.

2. Methodology for Lung Cancer Detection

This study employed lung histopathological images sourced from the LC25000 Dataset, which is available online. Andrew Borkowski and his colleagues from James Hospital Tampa, University of South Florida, and the Moffitt Cancer Center in Florida, USA, worked together to collectively assemble this dataset. The dataset encompasses histopathological images representing lung and colon cancer cases. Excluding colon cancer cases, the collection includes a total of 500 lung tissue images, divided equally between Benign Lung tissue and Lung Adenocarcinomas. These images were originally captured from pathology glass slides and were later resized to square dimensions of 768×768 pixels, down from their original size of 1024×768 pixels. The dataset underwent augmentation, resulting in an expansion to a comprehensive set of 10,000 lung histopathological color images which are categorized into two classes: Benign (N) and Adenocarcinoma (ACA), each consisting of 5000 images. These images are resized to a standard size of 256×256 followed by converting into a grey scale image. Notably, the images portray lung benign tissue characterized by abnormality but not indicative of cancer, while lung adenocarcinoma, the most prevalent form of lung cancer in the United States and notably linked to smoking, forms the second category.

Figure 1 shows the general schematic diagram for identifying and categorizing lung cancer in histopathological images. The input histopathological image will undergo conversion into a linear vector comprising 65,536 elements (due to the image's size of [256×256]). The procedure involves image pre-processing and a modified KFCM-based segmentation. During segmentation process approximately as [190×190] of the original image (i.e., nearly 36,100 intensity values) are segmented and used for further processing. These values will be directly employed to initialize the positions of birds in the Particle Swarm Optimization (PSO) and grey wolves in the Grey Wolf Optimization (GWO) algorithms. Optimization algorithms such as PSO and GWO are used to obtain a matrix of [512×10] dimensionally reduced intensity values from the segmented images. These dimensionally reduced features undergo feature selection techniques such as KL divergence and IWO. The selected features are then inputted into classifiers to evaluate their performance of the classifiers. Furthermore, an enhancement in the accuracy of lung cancer classification across various classifiers including SVM, KNN, Random Forest, Decision Tree, Softmax Discriminant,

Multilayer Perceptron, and BLDC classifiers is achieved through the implementation of a Hyper Parameter Updation algorithm based on the RAdam technique.
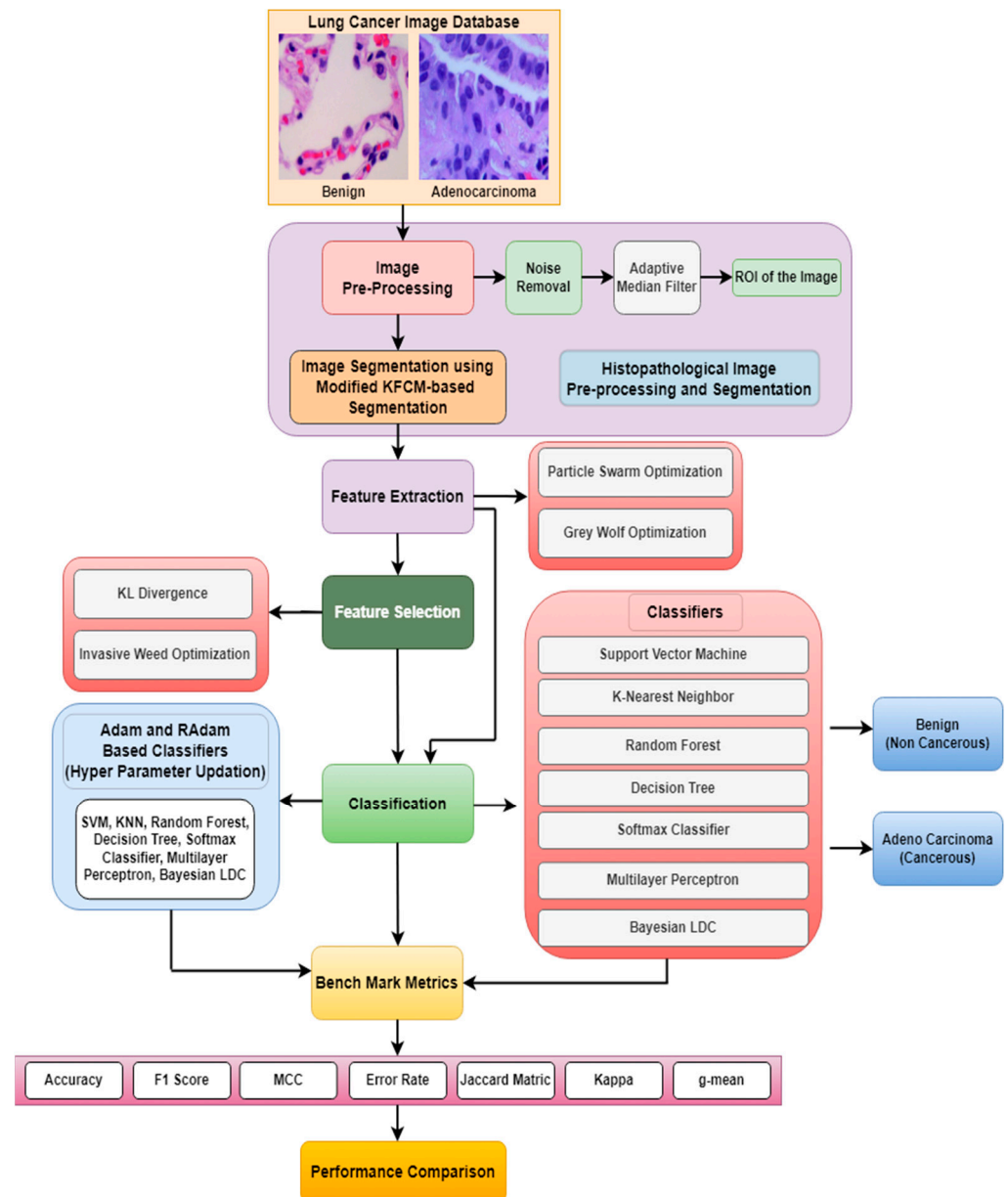
Figure 1. Schematic representation for detecting lung abnormalities from Histopathological Images.

2.1. Histopathological Image Preprocessing

Histopathological analysis serves as the definitive standard for evaluating the quality and clinical staging of tumors [35]. In the realm of diagnosing and treating medical conditions, healthcare professionals heavily rely on histopathological images. These images establish a crucial cornerstone for predicting patient survival rates [36].

As per available reports, histopathological images present several challenges:

- The images exhibit intricate geometric structures and complex textures that arise from the vast diversity in structural morphology [37].
- Notably, histopathological images are susceptible to color inconsistencies and noise due to external factors such as variations in illumination conditions [38].

- Variations in microscope magnification, equipment settings, and other variables contribute to inconsistencies in image sizes and resolutions within histopathological images [39].
- Elements of significance, such as local micro-vessels with distinctive textural characteristics, significantly influence disease diagnosis within histopathological images. Extracting these features is of paramount importance in supporting the classification and diagnosis of lung cancer [40].

Due to these factors, the histopathological images we encounter are frequently not perfect and these images show that image quality is affected by noise during acquisition and artifacts during sample preparation and slide digitization. Preprocessing methods are employed in histopathological images to enhance image quality, rectify anomalies, amplify pertinent characteristics, and establish uniformity, ultimately resulting in heightened precision and dependability of diagnostic outcomes. The study demonstrates that using an efficient adaptive median filter enhances image quality, reduces artifacts, and facilitates accurate diagnosis and analysis. However, when subjected to an adaptive median filter, these images tend to become smoother and exhibit reduced noise, rendering them suitable for our forthcoming investigations. After artifact removal, the filtered histopathological images are used for segmentation. Here, the size of the selected region of interest (ROI) is 256×256 which is the complete original image.

2.2. Histopathological Image Segmentation

A Modified Kernel Fuzzy C-Means methodology is employed to effectively segment normal and abnormal regions in histopathological images even though outliers are encountered. Image segmentation is the process of dividing an image into distinct regions based on certain image characteristics, with the goal of isolating and identifying specific regions within the image [41]. In this scenario, we have an input histopathological image denoted as H, which consists of a set of color images x_i at pixel i ($i = 1, 2, \ldots N$) and these color images are represented as $X = \{x_1, x_2, \ldots, x_N\} \subset R^k$, in the k-dimensional space. The cluster centers within the histopathological images are represented as $Y = \{y_1, y_2, \ldots, y_c\}$, where c is a positive integer ($2 < c \ll N$), and m_{ij} represents the membership value for each pixel i in the j-th cluster ($j = 1, 2, \ldots c$). In the Kernel Fuzzy C-Means algorithm, clusters are formed in the image space by assigning distinct membership values to all pixels. The objective function or general equation for the Kernel Fuzzy C-Means algorithm is expressed as follows in the Equation (1):

$$O_{KFCM} = \sum_{i=1}^{N} \sum_{j=1}^{c} m_{ij}^n \|x_i - y_j\|^2, \ 1 \leq n \leq \infty \tag{1}$$

where n represent an exponent used for regularization, with the condition that, $n > 1$, and $\|x_i - y_j\|^2$ denotes the squared grayscale Euclidean distance between x_i and y_j, which is given in Equation (2):

$$\sum_{j=1}^{c} m_{ij} = 1, \ m_{ij} \in [0,1], \ 0 \leq \sum_{i=1}^{N} m_{ij} \leq N \tag{2}$$

Using the membership function derived from the alternate optimization approach, the process of iteratively updating the cluster centers is carried out according to the Equations (3) and (4).

$$m_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\|x_i - y_j\|^2 / \|x_i - y_k\|^2 \right)^{1/(n-1)}} \tag{3}$$

$$y_j = \frac{\sum_{i=1}^{N} m_{ij}^n x_i}{\sum_{i=1}^{N} m_{ij}^n} \tag{4}$$

To reduce the impact of noise, the Equation (5) incorporates the spatial information of neighboring pixels,

$$O_{KFCM-S} = \sum_{i=1}^{N}\sum_{j=1}^{c} m_{ij}^n \|x_i - y_j\|^2 + \frac{\alpha}{N_R}\sum_{i=1}^{N}\sum_{j=1}^{c} m_{ij}^n \left(\sum_{r\in N_i} \|x_i - y_j\|^2\right) \tag{5}$$

Here, the spatial information is denoted by α, N_i represents the set of pixels and its cardinality is defined as N_R, the neighborhood function is substituted by $\|\acute{x}_i - y_j\|^2$, in place of $\frac{1}{N_R}\sum_{r\in N_R} \|x_i - y_j\|^2$, where, $\hat{x}$ represents a color scale-filtered image, and the Euclidean distance is replaced with the correlation distance measure to avoid the neighborhood function. The updated equation is represented in Equation (6):

$$O_{KFCM-S(1,2)} = \sum_{i=1}^{N}\sum_{j=1}^{c} m_{ij}^n \|x_i - y_j\|^2 + \alpha \sum_{i=1}^{N}\sum_{j=1}^{c} m_{ij}^n \left(\|x_i' - y_j\|^2\right) \tag{6}$$

In this study, a modified version of KFCM computes the parameter η_j for each cluster at every iteration to substitute for α [42]. The calculation of this parameter utilizes the correlation function, as outlined in the Equation (7):

$$\eta_j = \frac{\min_{j'\neq j}\left(1 - C\left(y'_j, y_j\right)\right)}{\max_k\left(1 - C(y_k, x')\right)} \tag{7}$$

Here C represents the correlation function or correlation distance measure. Here, determining the precise characteristics of C typically necessitates a large number of patterns and numerous cluster centers to identify optimal value for η_j. To address this challenge, a solution is devised by integrating spatial context and scale information through the incorporation of fuzzy factor. The objective function of the KFCM, as presented in Equation (8), incorporates the inclusion of the fuzzy factor F_{ij}.

$$O_{C-KFCM} = \sum_{i=1}^{N}\sum_{j=1}^{c}\left[m\left|ij^n\right| \|x_i - y_j\|^2 + F_{ij}\right] \tag{8}$$

Then the altered fuzzy factor F'_{ij} is derived using Equation (9).

$$F'_{ij} = \sum_{C\in N, i\neq k} w_{ik}\left(1 - m_{ij}\right)^m \tag{9}$$

This adjusted fuzzy factor plays a crucial role in influencing local neighbor relationships and substituting the traditional distance metric with a correlation function. Here, w_{ik} represents the fuzzy factor for cluster i, and $1 - C(x_i - y_j)$ signifies the correlation metric function. Since the histopathological images contain variation in intensities, gradients, and complex backgrounds, it becomes imperative to employ a modified KFCM-based segmentation method to distinguish between the region of interest (ROI) and the background in the image. Figure 2 illustrates the sequence of the original image, the filtered image, the identified ROI within the ACA image, and the segmented image generated using the modified KFCM for the Adenocarcinoma (ACA) class.
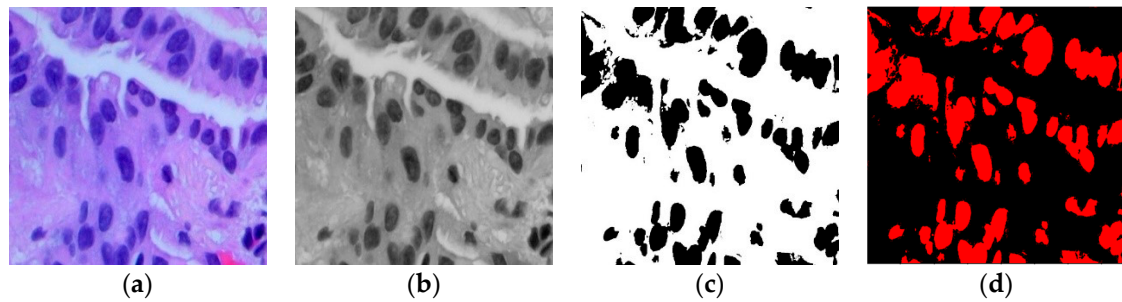
Figure 2. (**a**) Original ACA image; (**b**) Filtered ACA image; (**c**) ROI of the ACA Image; (**d**) Segmented ACA image.

The following section focuses on the methods utilized for extracting dimensionally reduced image features, aimed at enhancing the classification and detection of lung cancer using histopathological images.

3. Feature Extraction

Feature extraction techniques condense essential information from images into compact feature vectors, enabling the effective classification of complex image datasets using linear algorithms [43]. As the abundant features within histopathological images serve as a fundamental resource for clinicians to conduct diagnoses, the proficient extraction of these image features stands as a pivotal factor in enhancing the precision of computer-aided diagnosis [44]. This study delves into the impact of two distinct feature extraction techniques such as PSO and GWO on the classification of histopathological images related to lung cancer.

3.1. Particle Swarm Optimization (PSO)

Kennedy and Eberhart introduced the PSO algorithm in 1995, which draws inspiration from the hunting behavior of birds. This optimization method relies on a population and leverages the social dynamics of bird flocks. It starts by creating particles and setting key parameters for the optimization process. [45].

Every particle has a unique position that is traced by the following equation:

$$x_i^k = \left(x_{i1}^k, x_{i2}^k, \ldots, x_{iq}^k \right) \tag{10}$$

The velocity is traced by the following equation:

$$y_i^k = \left(y_{i1}^k, y_{i2}^k, \ldots, y_{iq}^k \right) \tag{11}$$

Each particle's velocity is updated as:

$$y_i^{k+1} = w_i y_i^k + c_1 r_1 \left(pbest_i - x_i^k \right) + c_2 r_2 \left(gbest_i - x_i^k \right) \tag{12}$$

Here, r_1 and r_2 represent randomly selected values within the range of 0 to 1. The acceleration coefficients, denoted as c_1 and c_2, play a role in analyzing the motion of particles. The weight function is expressed as:

$$w_i = \frac{w_{max} - w_{min}}{k_{max}} \times k \tag{13}$$

The position of each particle is given by:

$$x_i^{k+1} = x_i^k + y_i^{k+1} \tag{14}$$

The particle that possesses the optimal position progresses to the next level. The best position for an individual particle is represented by the letters "p-best", while the letters "g-best" represent the best position among all particles. The weight parameter "w_i" is chosen between 0.45–0.9, maximum iteration values are 100–1000, both r_1 and r_2 are set to 0.85, cognitive component (c_1) and Social Component (c_2) are chosen between 1.0–2.0. The above values are determined based on the trial-and-error method.

3.2. Grey Wolf Optimization (GWO)

Grey wolves are known for living and hunting in groups called packs [46]. The process of searching and hunting involves plotting to track and approach a target efficiently. This optimization technique, inspired by the search and hunting patterns of gray wolves, employs symbols such as Alpha (α), Beta (β), and Gamma (γ) to represent the best, next best, and third best solutions in mathematical modeling. Lambdas are presumed to be the remaining possible solutions and they guide the alpha, beta, and gamma wolves in searching and surrounding the prey. Three coefficients, A, B, and C are suggested to describe the encircling behavior. The equation of hunting strategy is formulated as follows:

$$D_\alpha = |B_1 \cdot X_\alpha - X(t)| \tag{15}$$

$$D_\beta = |B_2 \cdot X_\beta - X(t)| \tag{16}$$

$$D_\gamma = |B_3 \cdot X_\gamma - X(t)| \tag{17}$$

where D_α, D_β and D_γ denotes the adjusted distance variables from the alpha, beta, and delta positions to the other wolves, B_1, B_2 and B_3 are coefficients that assist in adapting these distance variables, t signifies the ongoing iteration, X indicates the position of the grey wolf and it follows as,

$$X_1 = |X_\alpha - A_1 D_\alpha| \tag{18}$$

$$X_2 = |X_\beta - A_2 D_\beta| \tag{19}$$

$$X_3 = |X_\gamma - A_3 D_\gamma| \tag{20}$$

$$X(t) = \frac{X_1 + X_2 + X_3}{3} \tag{21}$$

The parameters A and B can be mathematically expressed as follows:

$$A = 2i \cdot r_1 - i \tag{22}$$

$$B = 2 \cdot r_2 \tag{23}$$

The control parameter i chases A, which eventually drives the lambda wolves to flee from the dominant wolves such as α, β and γ. When there are multiple dominant wolves ($|A| > 1$), the grey wolves run away from them, allowing lambda wolves to search extensively and explore more during optimization. However, when there are fewer dominant wolves ($|A| < 1$), the grey wolves approach them and follow their guidance in hunting, which is called local search in optimization. During the iterations, the control parameter i is linearly decreased from 2 to 0, and is represented as,

$$i = 2 - (iter) \cdot \frac{2}{max\_iter} \tag{24}$$

where max_iter indicates the maximum iteration, and it is started from the beginning.

In the context of the classification problem, the introduction of randomness through variables r_1 and r_2 leads to heightened fluctuations in the wolves' positions. Consequently, their ability to effectively converge towards the target (prey) becomes hindered. To address

this issue, a decision has been made to treat the values of r_1 and r_2 in Equations (9) and (10) as control parameters within a confined range of [0, 1], rather than allowing them to remain purely random. Through empirical experimentation, it has been determined that the optimal performance of the Grey Wolf Optimization (GWO) algorithm is achieved when both r_1 and r_2 are set to 0.8. This adjustment enhances the accuracy of the GWO algorithm in tackling the classification problem.

3.3. Statistical Analysis

To enhance the accuracy of cancer prediction using dimensionally reduced features, it is advisable to calculate statistical parameters from the region of interest. The intensity values, which have been reduced in dimensionality through methods such as PSO (Particle Swarm Optimization) and GWO (Grey Wolf Optimization), are then examined using statistical measures such as Mean, Variance, Skewness, Kurtosis, Pearson Correlation Coefficient (PCC), and CCA (Canonical Correlation Analysis). These statistical parameters help determine whether the outcomes accurately reflect the inherent properties of lung cancer data within the subspace. These attributes were derived for both normal and malignant classes.

The statistical parameters of cancer data, extracted using the PSO and GWO methods, are shown in Table 1. Variance quantifies data spread. Notably, Table 1 reveals lower mean values for normal cases using both PSO and GWO, while higher mean values are evident for malignant cases using both methods. Furthermore, the Malignant group demonstrates greater data spread compared to the Normal group as indicated by Table 1. GWO shows a Pearson correlation coefficient of 1 for both cases, implying strong intra-class correlation. Skewness and kurtosis are highly skewed for both normal and malignant instances. When CCA values exceed 0.5, strong inter-class correlation is present. However, Table 1 indicates that PSO and GWO methods exhibit the lowest inter-class correlation. Consequently, the analysis of these extracted features emphasizes the need for improved classifiers.

Table 1. Statistical Parameters in PSO and GWO for Feature Extraction in Malignant and Normal Data.

Statistical Parameters	PSO		GWO	
	Malignant	Normal	Malignant	Normal
Mean	0.8598080214	0.1109701363	0.01878313748	0.01751341349
Variance	0.05867975074	0.07425036326	0.07492946326	0.07494543857
Skewness	19.87029488	19.83047771	22.52231557	22.56212107
Kurtosis	441.8828416	444.9961882	509.1565306	510.3537192
Pearson CC	0.9019022281	0.9269991469	0.9985202125	0.997858273
CCA	0.12309		0.11291	

In cases where the features exhibit linear separability, a straightforward binary thresholding approach can be employed for the classification of Histopathological Lung images into two distinct classes: N and ACA. The characteristics of malignancy exhibit non-linear and non-Gaussian features that overlap with each other. To analyze these dimensionally reduced values which was obtained from PSO and GWO methods, histogram and scatterplot plots are used as illustrated in Figures 3 and 4.
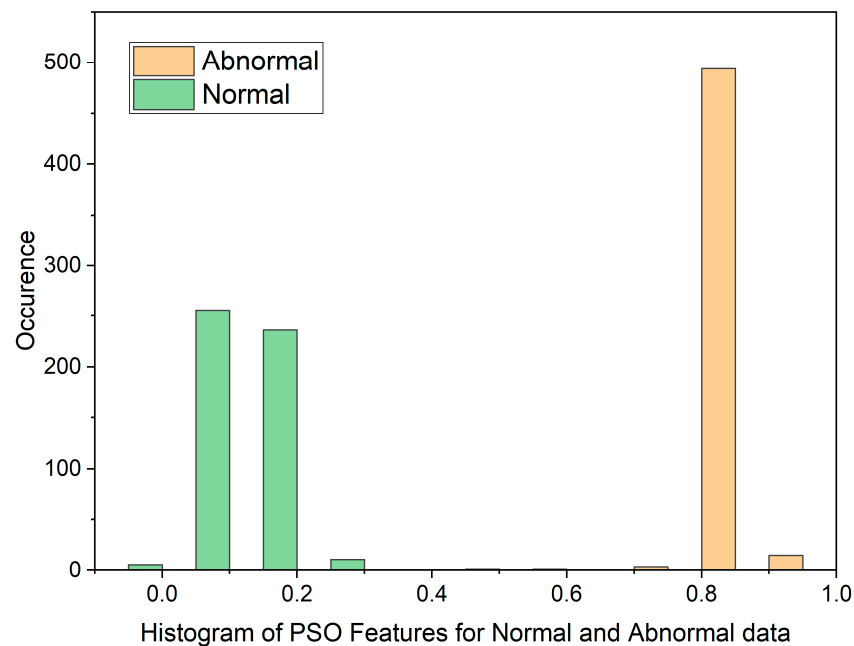
Figure 3. Histogram of PSO Features for Normal and Malignant data.
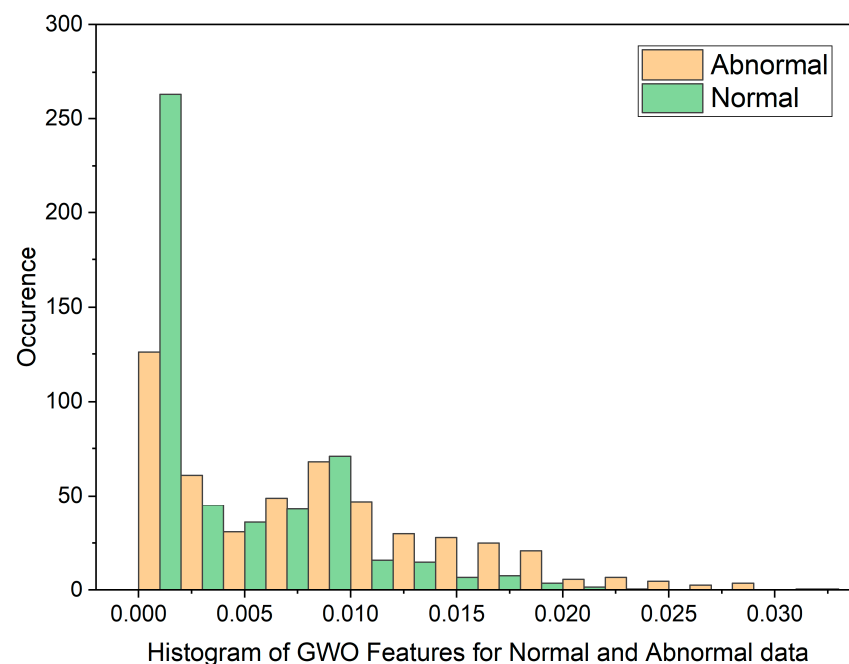
Figure 4. Histogram of GWO Features for Normal and Malignant data.

The histogram plot in Figure 3 illustrates the distribution of PSO feature data for normal and malignant cancer cases. The histogram illustrates PSO features characterized by outliers, substantial gaps, downward trends, and a non-Gaussian distribution. From Table 1, In the PSO-based extraction technique, the Canonical Correlation Coefficient (CCA) value is significantly low at 0.12309, suggesting a non-linear relationship between normal and malignant cases. Figure 4 showcases the histogram plot for GWO feature distribution, indicating skewed Poisson distributed data, and a non-linear nature.

Figures 5 and 6 display scatterplots demonstrating the feature output of normal and malignant cancer data utilizing the PSO and GWO methods. Scatter plots are useful for identifying data clustering, detecting nonlinearity, and overlapping. Both figures indicate

the presence of nonlinearity and overlapping in the data. Therefore, from the histogram and scatterplot it is evident to employ accurate classifiers capable of distinguishing between normal and cancer cases in lung data using PSO and GWO features. The next section centers on the techniques applied to choose optimal image features, with the goal of improving the classification and identification of lung cancer in histopathological images.
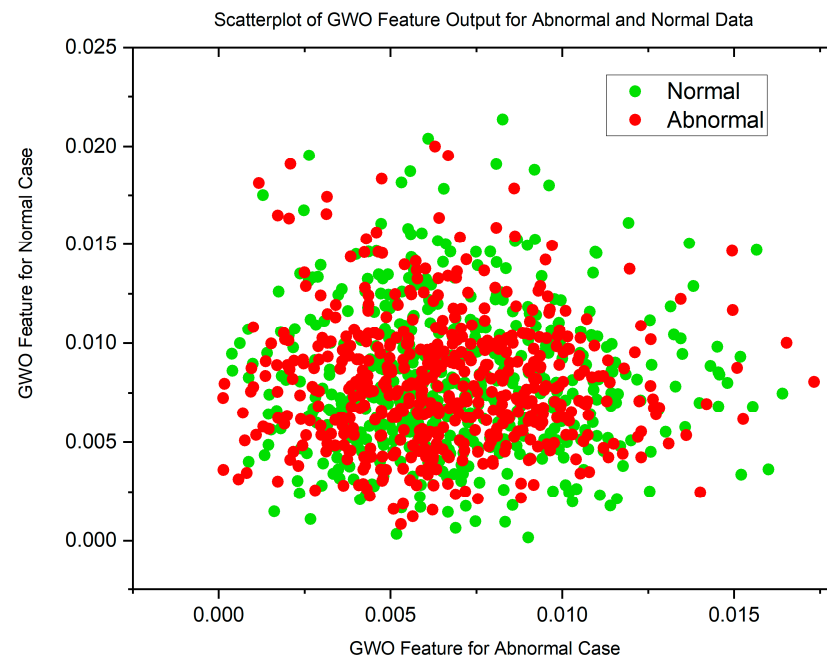
Figure 5. Scatterplot of PSO Features for Normal and Malignant Case.
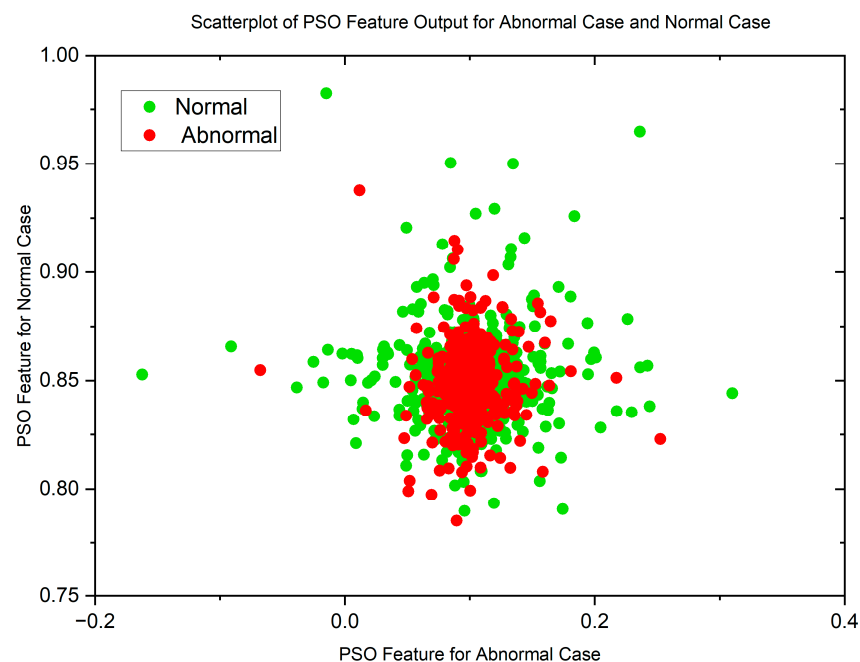
Figure 6. Scatterplot of GWO Features for Normal and Malignant Case.

4. Feature Selection

Feature selection aims to reduce input variables, excluding irrelevant characteristics for a more accurate, less complex, and unbiased model. Optimal feature selection is crucial for creating an effective, accurate machine learning model with high generalization ability [40]. In this paper, Feature Selection is performed using the KL Divergence and Invasive Weed

Optimization (IWO) methods. Following the feature extraction procedures as described in Section 3, which involves Particle Swarm Optimization (PSO) and Grey Wolf Optimization (GWO). After feature extraction method [256 × 256] is dimensionally reduced to [512 × 10] per histopathological image. These [512 × 10] intensity values per image serve as the initial input for the Feature Selection techniques namely KL Divergence and IWO. However, the application of Feature Selection techniques is a further dimensionality reduction method only. After PSO and GWO feature extraction methods through KL Divergence and IWO process [512 × 10] intensity values per image is reduced to [100 × 12] which represents the most relevant intensity values of the images are retained as input given to the classifier for the subsequent classification process. The histopathological images are represented as a relevant intensity value in matrix form as described above.

4.1. KL Divergence

KL Divergence, also known as relative entropy, measures disparities between probability distributions, but in an asymmetric manner. The KL divergence between a probability distribution $q = (q_1, q_2, \ldots, q_n)$ and another distribution $p = (p_1, p_2, \ldots, p_n)$ is defined as,

$$D_{KL}(q \mid \mid p) = \sum_{j=1}^{m} q_j log \frac{q_j}{p_j} \tag{25}$$

The integral form of the *KL* divergence for continuous distributions is expressed as follows:

$$D_{KL}(q \mid \mid p) = \int_{-\infty}^{\infty} q_j log \frac{q_j}{p_j} dx \tag{26}$$

The *KL* divergence exhibits mutual convexity for both discrete and continuous distributions. The following are the properties of the *KL* divergence measure:

$$D_{KL}(q \mid \mid p) = \begin{cases} 0, \; if \; q = p \\ c, \; c > 0, \; if \; partially \; overlapping \\ +\infty, \; if \; non-overlapping \end{cases} \tag{27}$$

From the above equation, it can be observed that when the *KL* divergence is smaller, the two compared distributions are more similar.

4.2. Invasive Weed Optimization

The invasive weed optimization algorithm is a popular population-based metaheuristic approach [47]. The dynamic and versatile characteristics of weed colonies have sparked the creation of an optimization algorithm that imitates their behavior. By leveraging the qualities of weeds, a straightforward and efficient optimization technique can be developed. This method, called the IWO algorithm, incorporates phases such as seeding, growth, and competition. The following are the strategy for simulating weed habitat behavior:

1. Primary Population Initialization: A few seeds are dispersed to start the search.
2. Reproduction process: Seeds have the potential to grow into flowering plants, which then choose and spread the fittest seeds for survival and reproduction. The quantity of grass grain grains decreases in a linear fashion from Y_{max} to Y_{min} as follows:

$$n(weed_j) = \frac{Y_{\max}(\max\_fit - fit(weed_j)) + Y_{\min}(fit(weed_j) - \min\_fit)}{\max\_fit - \min\_fit} \tag{28}$$

3. Spectral Spread Method: The group's seeds are distributed normally with a mean planting position and standard deviation (SD) determined by the equation below.

$$\sigma_t = \left(\frac{N-t}{N}\right)^m \left(\sigma_{intital} - \sigma_{final}\right) + \sigma_{final} \tag{29}$$

4. Competitive Deprivation: If the colony has more grasses than the maximum limit (Smax), the grass with the lowest fitness is eliminated to maintain a consistent number of herbs.
5. The process continues until the maximum iteration is reached, keeping the lowest cost value of the grasses.

The upcoming next section revolves around the utilization of classification methods to categorize lung cancer images within histopathological images.

5. Classifiers for the Detection of Lung Cancer

Classifiers have a crucial role in categorizing data effectively. An optimal classifier is characterized by its ability to achieve high accuracy and low error rates while maintaining manageable computational complexity. Addressing the classification challenge involves constructing a model for the purpose of classifying images and assigning them appropriate class labels. The following sections of this paper delve into the classifiers that were used for this purpose.

5.1. Support Vector Machine

SVM is known for its scalability and classification performance [42]. It aims to create a hyperplane that maximizes class separation by minimizing the cost function. It is given by the following expression:

$$Minimize, \frac{1}{2}\|w\|^2 + C\sum_{k=1}^{m}\mu_k \tag{30}$$

$$\text{Subject to } z_k\left(w^T x_k + f\right)^3 \geq 1 - \mu_k, \mu_k \geq 0$$

where $w^T, x_k \in R^2$ and $f \in R, \|w\|^2 = w^T w$.

C represents the trade-off between the margin and the error. The training data's size is represented by $\sum_k$, and the class label for each sample is represented as z_k. SVM is a flexible classifier suitable for linear and nonlinear cases. To handle nonlinear data, we employ Polynomial, RBF, and Sigmoid kernel functions. In this study, we exclusively enhance the classification accuracy by utilizing the SVM-RBF kernel.

5.2. K-Nearest Neighbor

KNN stands as a widely utilized and efficient non-parametric classification technique. In KNN, the symbol 'k' denotes the count of nearest neighbors involved in the voting process. To enhance prediction accuracy, employing an odd value for k is recommended. KNN determines the classification of a test sample by conducting a majority vote among neighboring training samples. Measuring distances between individuals is crucial, and the Euclidean distance is commonly used for this purpose [48]. For example, in the Euclidean space if u and v are the two points and it is assumed that $u = (u_1, u_2, u_3, \ldots, u_n)$ and $v = (v_1, v_2, v_3, \ldots, v_n)$, then the Euclidean distance of line segment can be expressed as follows:

$$Dist(u, y) = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2 + \ldots + (u_n - v_n)^2} = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2} \tag{31}$$

5.3. Random Forest

This tree-based ensemble learning algorithm is highly accurate and resilient in image classification [49]. It utilizes multiple decision trees that work independently. Two important parameters for the algorithm are the number of decision trees and the number of predictive variables used in each tree's decision-making process. By combining the votes of multiple decision trees, a random forest can accurately predict binary tasks. For a training set $\times$ consisting of M samples, each containing N features and a classification label Y. The following steps are involved in the construction of Random Forest.

1. Randomly select M samples from × using the Bootstrap method.
2. Choose *n* random features (where *n* < N) to split a decision tree node. Determine the split criterion by selecting the feature with the lowest Gini value. Gini is computed using the formula:

$$Gini = 1 - \sum_{i=1}^{c} \left(p_i \right)^2 \tag{32}$$

where p_i represents the relative frequency of dataset features and *c* represents the number of classes.
3. Generate M decision trees by repeating steps 1 and 2, M times.
4. Create a random forest by combining the decision trees and utilize voting to determine the classification outcome.

5.4. Decision Tree

It is a well-known machine learning algorithm that partitions input data recursively [50]. A decision tree starts with a root node and branches. This work utilizes CART, which splits the data based on its ability to distinguish between groups. The process continues until all data groups have the same label or match the training set. CART uses the Gini impurity measure at each node to determine the best split. The data at node 'd' are divided into two subsets, X-left and X-right, based on the splitting features and a threshold determined by CART and the amount of data X.

At node 'd' the input is computed through impurity measure Gini as $\sum_k p_{dk}(1 - p_{dk})$ with the proportion of class k observation in the node 'd'. Construction time of a decision tree depends on the dataset's size (samples and features). Overfitting can occur if the tree is built using CART and results in few samples per leaf. To prevent overfitting and improve accuracy, a pruning algorithm can be used to simplify the tree, reducing construction time while maintaining performance.

5.5. Softmax Discriminant Classifier

SDC's main objective is to classify a given test specimen [51] by comparing its distance to the training sample within its category. The process entails gauging the distance between training and test samples belonging to the same class to derive the outcome. Supposing, the training set $M = [M1, M2, \ldots, Mq] \in \mathfrak{R}^{c \times d}$ comes from q distinct classes. $M_q = [M1^q, M2^q, \ldots Mdq^q] \in \mathfrak{R}^{c \times d_q}$ Indicates *dq* samples from the qth class where $\sum_{j=1}^{q} d_j = d$. Assuming $w \epsilon \mathfrak{R}^{c \times 1}$ represents the test sample, within the classifier, we employ samples from class q to recognize the test sample, aiming to minimize the reconstruction error. To uphold the principle of SDC, we can enhance the non-linear transformation linking the q class samples and the test sample. Therefore, the SDC can be defined as follows:

$$h(w) = \underset{j}{\operatorname{argmax}} z_w^j \tag{33}$$

$$h(w) = \underset{j}{\operatorname{argmax}} \log \left(\sum_{j=1}^{d_j} \exp \left(-\lambda \left\| w - w_k^j \right\|_2 \right) \right) \tag{34}$$

where $h(w)$ defines the distance between the jth class and the test sample. The value of λ should be greater than zero, to provide a penalty cost. If w relates to the jth class, then w and w_k^j would have likely same characteristics and so $\left\| w - w_k^j \right\|_2$ is progressing close towards zero and hence maximizing z_w^j can achieve the maximum possible value in an asymptotic manner.

5.6. Multilayer Perceptron

MLP is often used to approximate functions such as regression [52]. It consists of an input layer with *n* nodes, a hidden layer, and an output layer. The given input and output

pairs be denoted as (m_p, n_p), $p = 1, 2, \ldots, m$, where $m_p = (m_{p1}, m_{p2}, \ldots, m_{pn})$ and y_p are the input vector and the corresponding desired output value, respectively. Sigmoid function is commonly used for hidden and output nodes, producing values from 0 to 1.

The kth hidden node in the MLP calculates its output when the input is given. The output value is computed as

$$c_{pk=} f_s \left(\sum_{j=1}^{n} w_{jk} m_{pj} + \theta_k \right) \tag{35}$$

The output value of the output node is determined by the sigmoid function (f_s), along with bias (θ_k), and connection weight (w_{jk}) associated with the corresponding hidden node. Then the final output value is computed as,

$$c_{p=} f_s \left(\sum_{k=1}^{l} w_k c_{pk} + \theta \right) \tag{36}$$

The number of hidden nodes is denoted by l, the bias to the output node is represented by θ, and w_k signifies the connection weight from the kth hidden node to the output node. This results in a total of $(n+2)l+1$ synaptic connections. To train the Multilayer Perceptron (MLP), the following cost function can be utilized.

$$E = \frac{1}{2} \sum_{j=1}^{t} (n_p - c_p)^2 \tag{37}$$

where t denotes the number of training patterns. In our study, we used a three-layer model, which is known to effectively approximate any continuous function with high accuracy [53].

5.7. Bayesian Linear Discriminant Classifier

The BLDC, or Bayesian Linear Discriminant Classifier, can distinguish between multiple classes. It uses the Fisher linear discriminant and applies the Bayes decision rule to estimate the error probability [54]. Bayesian regression assumes that the target variable y is a linear combination of vector k, and Gaussian noise m. This relationship is expressed as $y = q^T k + m$, where q represents the weight coefficients.

The given expression represents the likelihood function,

$$p \left(\frac{C}{\beta, q} \right) = \left(\frac{\beta}{2\pi} \right)^{m/2} \exp \left(-\frac{\beta}{2} \left\| M^T q - y \right\|^2 \right) \tag{38}$$

In the above equation, y is the target values for regression, M is a matrix made by combining the training feature vectors horizontally, and C is the combination of $\{M, y\}$. β represents the noise's inverse variance, and T is the total number of samples in the training set.

5.8. Methods for Updating Hyperparameters in Various Classifiers

The performance of a classifier greatly depends on the values assigned to its hyperparameters [55]. To find the best hyperparameter values, different methods such as Stochastic Gradient Descent (SGD), Grid Search (GS), and Adaptive Moment Estimation Method (ADAM) can be used. This study introduces a new approach called R-Adam, which aims to enhance lung cancer classification accuracy for the Decision Tree classifier and other classifiers. While Adam is a prevalent choice for hyperparameter selection in deep learning networks, this study introduces R-Adam, an adapted version proposed for hyperparameter selection across diverse classifiers. Utilizing controlled randomness, the envisioned R-Adam algorithm aims to discover hyperparameter values in proximity to the optimal

values recommended by the Adam method. The investigation assesses the classification performance using both Adam and the newly introduced R-Adam technique.

5.8.1. Adam Approach

The Adam approach involves employing squared gradients and exponential moving averages. The validation of hyperparameters is achieved based on the expressions provided below [56]:

$$x_{t+1} = x_t - \frac{L_r}{\varepsilon + \sqrt{\hat{P}_t}} * \hat{M}_t \tag{39}$$

where x_t represents the previous hyperparameters, x_{t+1} denotes the updated hyperparameters, L_r signifies the learning rate, and ε is a small constant used to avoid division by zero. The constants in the Adam method are Z_1 and Z_2.

$$\hat{M}_t = \frac{m_t}{1 - Z_1^t} \tag{40}$$

$$\hat{P}_t = \frac{p_t}{1 - Z_2^t} \tag{41}$$

$$m_t = Z_1 * m_{t-1} + (1 - Z_1) * \frac{\partial L}{\partial x_t} \tag{42}$$

$$p_t = Z_2 * p_{t-1} + (1 - Z_2) * \left(\frac{\partial L}{\partial x_t} \right)^2 \tag{43}$$

where $\frac{\partial L}{\partial x_t}$ signifies the derivative of the loss function with respect to x. Thus, the mathematical representation of the loss function is as follows:

$$\frac{\partial L}{\partial x_{tr}} = \frac{ER_{tr}}{x_{in}}, \; if \; tr = 1 \tag{44}$$

$$\frac{\partial L}{\partial x_{tr}} = \frac{ER_{tr} - ER_{tr-1}}{x_{tr} - x_{tr-1}}, \; if \; tr > 1 \tag{45}$$

where *ER* stands for the error rate, *tr* indicates the current iteration and $tr - 1$ denotes the previous iteration of in the Adam approach. Algorithm 1 outlines the process of utilizing the Adam optimizer to update hyperparameters in a Decision Tree model, aiming to minimize the error rate, which serves as the loss function. In Decision Tree, the key hyperparameters include maximum depth and criterion. In Decision tree, the hyperparameters are set as maximum depth = 20 and criterion = MSE. The Adam's approach employs specific constants in this work: $L_r = 0.001$, $Z_1 = 0.89$, $Z_2 = 0.9$ and $\varepsilon = 10^{-9}$. Through experimentation, the optimal number of iterations for the Adam approach was determined to be 40. This iterative process aims to uncover the lowest error rate, helping identify the best hyperparameters. Notably, similar approaches involving SVM, KNN, Random Forest, SDC, MLP, and BLDC models could also leverage the Adam optimizer to update hyperparameters in a comparable manner.

Algorithm 1. Adam Approach

---

1. Initialization: Set initial values for hyperparameters:

    Target value, maximum iterations, maximum depth and criterion for Decision Tree, maximum iterations for Adam, L_r, Z_1, Z_2, ε.

2. Hyperparameter Tuning Loop:

    (a) For t_r = 1 to maximum iterations for Adam

    - Calculate maximum depth.
    - Determine the criterion.

    (b) For t = 1 to maximum iterations for Decision Tree

    - Update values for maximum depth
    - Criterion is set to MSE.
    - Determine the optimal values for maximum depth.

    end for.

3. Formulate a confusion matrix and compute the error rate (ER).
4. Compute the loss gradient using Equation (45).
5. Establish new optimal hyperparameter values using Equation (39) through (43).
    end for.

5.8.2. RAdam's Approach

The Randomized Adam (RAdam) technique is tailored to enhance the precision of the Decision Tree classifier. Algorithm 2 presents a methodology for implementing the Decision Tree using the RAdam approach. RAdam amalgamates two core components: the Adam method and controlled randomization. The controlled randomization process is pivotal in elevating classification performance. Within each iteration of the Adam method, hyperparameters are updated. The Adam process, which meticulously refines hyperparameter ranges, is nested within the iterative controlled randomization. This controlled randomization strategy integrates two control parameters—solution considering rate and solution adjusting rate—to fulfill its objective. Constants for R-Adam are defined as follows: bandwidth is set at 0.0098, the maximum number of iterations for randomization is 15, solution considering rate is 0.6, and solution adjusting rate is 0.92. In Algorithm 2, randomization 1, randomization 2, randomization 5, and randomization 6 indicate random values from the range [0, 1], while randomization 3 and randomization 4 correspond to random values within [0, 0.1]. Following this iterative process, the lowest error rate is found, leading to the identification of optimal hyperparameters. Significantly, analogous methodologies that pertain to SVM, KNN, Random Forest, SDC, MLP, and BLDC models could also make use of the Adam optimizer for adjusting hyperparameters in a similar fashion.

The following section pertains to the outcomes derived from employing diverse classification techniques for the categorization of lung cancer images within histopathological images.

Algorithm 2. RAdam's Approach

---

1. Initialization: Set initial values for hyperparameters:

    Target value, maximum iterations, maximum depth and criterion for Decision Tree, maximum iterations for Adam, L_r, Z_1, Z_2, ε, solution considering rate, solution adjusting rate and bandwidth.

2. Hyperparameter Tuning Loop:

    (a)   For t_r = 1 to maximum iterations for Adam

    - Calculate maximum depth.
    - Determine the criterion.

    (b)   For t = 1 to maximum iterations for Decision Tree

    - Update values for maximum depth
    - Criterion is set to MSE.
    - Determine the optimal values for maximum depth.

    end for.

3. Formulate a confusion matrix and compute the error rate (ER).
4. Compute the loss gradient using Equation (45).
5. Establish new optimal hyperparameter values using Equation (39) through (43).
    end for.
6. For each iteration: current iterations for randomization = 1 to maximum iterations for randomization
    If randomization 1 < solution considering rate.
    Set r_1 for this iteration as $r_1\prime$
    Set r_2 for this iteration as $r_2\prime$
    If randomization 2 < solution adjusting rate.
    Set r_1 for this iteration as $r_1\prime$+bandwidth$\times$ randomization 3.
    Set r_2 for this iteration as = $r_2\prime$+bandwidth$\times$ randomization 4.
    end if.
    If r_1 for this iteration is less than the lower bound, set it to the lower bound.
    end if.
    If r_2 for this iteration is less than the lower bound, set it to the lower bound.
    end if.
    If r_1 for this iteration is less than the upper bound, set it to the upper bound.
    end if.
    If r_2 for this iteration is less than the upper bound, set it to the upper bound.
    end if.
    Set r_1 for this iteration as lower bound + (bandwidth $\times$ randomization 5).
    Set r_2 for this iteration as lower bound + (bandwidth $\times$ randomization 6).
    end if.
7. Repeat.

    - Calculate maximum depth.
    - Determine the criterion.

8. For each iteration t from 1 to maximum iterations for Decision Tree

    - Update the values of maximum depth.
    - Set the criterion to MSE.
    - Determine the optimal values for maximum depth.

    end for.

9. Formulate a confusion matrix and compute the error rate (ER).
10. Compute the ER using r_1 & r_2 as hyperparameters. end for.

---

6. Results and Discussion

This section explores the efficacy of different classifiers based on their benchmark parameters. A higher classification accuracy combined with a decreased error rate signifies robust performance of the classifier. As a result, the classifiers underwent training and testing using the extracted and chosen feature values within the Lung Histopathological Image Dataset.

6.1. Training and Testing of the Classifiers

The training and testing of the classifiers constitute crucial phases within classification procedures. Training facilitates the acquisition of patterns linked to the provided dimensionally reduced intensity values of the histopathological images by the classifier. In this study, the entire dataset, comprising histopathological image values related to lung cancer detection and classification, is divided into 10 equal folds. The analysis involves a series of iterations. During each iteration, one-fold is designated as the testing set, while the remaining nine folds are combined and used as the training set. In essence, 10% of the data is reserved for testing in each iteration, and the remaining 90% is utilized for training. Various performance metrics are computed for each iteration. The results obtained from all 10 iterations are collected and aggregated. This aggregation often involves calculating average values for the performance metrics. The conclusion of training and testing for the classifiers was established based on the mean square error (MSE) acting as the termination criterion. The mathematical expression for MSE is given below:

$$MSE = \frac{1}{M} \sum_{i=1}^{M} (O_i - T_k)^2 \tag{46}$$

where O_i signifies the value observed at a definite time; T_k indicates the target value for model k, with "k" ranging from 1 to 15; and the value of M is assumed to be 5000 and indicates the total number of images.

6.2. Selection of the Optimal Parameters for the Classifiers

In this study, seven classifiers were used to categorize images into benign or adenocarcinoma based on the target selection. The target selection for the benign case $\left(T_{benign} \right)$ is represented as follows:

$$\frac{1}{M} \sum_{k=1}^{M} \mu_k \leq T_{benign} \tag{47}$$

The characteristics of the entire set of benign lung data (M) were subjected to normalization, and their average is denoted as μ_k as outlined in Equation (38), applicable for classification purposes.

The average of the normalized features is denoted as μ_k. For benign images, a target value of 0.1 was selected, which falls within the lower end of the 0–1 scale.

The condition for choosing a target in a case of adenocarcinoma (aca) is:

$$\frac{1}{N} \sum_{i=1}^{N} \mu_i \leq T_{aca} \tag{48}$$

The characteristics of the entire set of lung adenocarcinoma data (N) were subjected to normalization, and their average is denoted as μ_i as outlined in Equation (39), applicable for classification purposes.

To enhance adenocarcinoma classification, the target selection should exceed the mean value μ_k, which represents the average of normalized features across N images. Improving classification requires a target value of 0.5 or higher, as specified by the condition:

$$\left(\left\| T_{aca} - T_{benign} \right\| \geq 0.5 \right) \tag{49}$$

Depending on the criteria described in Equation (40), the selected targets for this study were set at 0.1 for benign cases and 0.85 for adenocarcinoma cases. The classifiers underwent training using a 10-fold cross validation training and testing approach, with the stopping criterion being an MSE value of $\left(10^{-5} \right)$ or a maximum operation of 1000, whichever was achieved first. The selection of optimal parameters for the classifiers during the training process is outlined in Table 2. In the case of SVM (RBF) classifier the parameters

are selected through the trial-and-error method. The classifiers parameters are α, Kernel width parameter (σ), w, and b are selected with the constraint of minimum MSE. In the case of KNN, K value indicates the number of clusters and in this case, it is K = 5 is selected randomly. With Euclidean Distance measure as the cluster coefficient with weight w = 0.5 is selected with the constrain of minimum MSE. In the case of Random Forest, the parameters such as Number of trees, Maximum depth and Bootstrap sample are initialized with random selection. Similarly in the case of Decision Tree, the parameter maximum depth is initialized with random selection. Since it is a binary classification problem, the class weight value for Random Forest is settled at 0.45, whereas for Decision tree, the class weight is settled at 0.4. In the case of Softmax Discriminant Classifier, it is a binary classification problem, so the λ value is settled at 0.5 along with the mean of each class target values as 0.1 and 0.85. In the case of Multilayer Perceptron Classifier, the network is trained using LM (Levenberg-Marquardt) algorithm to minimize the square output error. This error back propagation algorithm is used to calculate the weights updates in each layer of the network. As the number of hidden units gradually increased from its initial value, then there will be a reduction in the minimum Mean Squared Error (MSE) on the testing set. The optimal number of hidden units is the one that results in the lowest MSE. If the number of hidden units is increased beyond this point, the model's performance does not show any further improvement; instead, it often starts to decline. This decline occurs since the neural network becomes unnecessarily complex, exceeding the complexity necessary to solve the problem effectively. The choice of the learning rate as 0.3 is determined based on the distribution of training patterns and their associated MSE. In case of BLDC, the parameters such as prior probability p(x) − 0.5, Class mean $\mu_x = 0.8$ and $\mu_y = 0.1$ are selected with constrain of minimum MSE. The training process demonstrated that the MSE value was attained either as low as 1.0×10^{-10} or after 1000 iterations.

Table 2. Selection of the Optimal Parameters for the Classifiers.

Classifiers	Optimal Parameters of the Classifiers
Support Vector Machine	Kernel—RBF; α—1; Kernel width parameter (σ)—100; w—0.85; b—0.01; Convergence Criterion—MSE.
K-Nearest Neighbor	K—5; Distance Metric—Euclidian; w—0.5; Criterion—MSE.
Random Forest	Number of Trees—200; Maximum Depth—10; Bootstrap Sample—20; Class Weight—0.45.
Decision Tree	Maximum Depth—20; Impurity Criterion—MSE; Class Weight—0.4.
Softmax Discriminant Classifier	λ = 0.5 along with mean of each class target values as 0.1 and 0.85.
Multilayer Perceptron	Learning rate—0.3; Learning Algorithm—LM; Criterion—MSE.
Bayesian Linear Discriminant Classifier	Prior Probability P(x)—0.5; Class mean $\mu_x = 0.8$ and $\mu_y = 0.1$, Criterion = MSE.

6.3. Performance Metrics of the Classifiers

The primary objective of the classifier was to effectively distinguish between cancer cells and normal data samples in the dataset. As this research focuses on binary classification, it is essential to select appropriate performance metrics. In binary classification tasks, one of the key evaluation tools is the confusion matrix. This matrix provides a concise summary of the model's predictions in relation to the actual labels of the dataset. The confusion matrix consists of four elements: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP indicates the presence of lung cancer, while TN indicates its absence, both representing correct classification. FP and FN represent misclassification, where lung cancer is incorrectly predicted as present (FP), or lung cancer is present but wrongly classified as not present (FN).

Table 3 displays TP, TN, FP, FN values, and average MSE for PSO and GWO features along with seven classifiers without employing Feature Selection Methods. Achieving the

lowest MSE serves as an indicator for improved classifier performance, while a higher MSE value results in inferior classifier performance, regardless of the employed feature selection methods. PSO features show Decision Tree Classifier with the lowest MSE (3.60×10^{-7}) and Random Forest Classifier with the highest MSE (1.60×10^{-5}). GWO features show Bayesian LDC Classifier with the minimum MSE (2.50×10^{-7}) and KNN Classifier with the maximum MSE (1.44×10^{-5}).

Table 3. Confusion Matrix for Classifiers without Feature Selection.

Feature Extraction	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
PSO	SVM	3944	4009	991	1056	7.29×10^{-6}
	KNN	4267	3725	1275	733	4.49×10^{-5}
	Random Forest	2692	2933	2067	2308	1.60×10^{-5}
	Decision Tree	3184	3217	1783	1816	3.60×10^{-7}
	Softmax Discriminant	4033	3750	1250	967	4.00×10^{-8}
	Multilayer Perceptron	3425	3675	1325	1575	2.25×10^{-6}
	Bayesian LDC	4367	3975	1025	633	5.63×10^{-5}
GWO	SVM	3617	4175	825	1383	5.76×10^{-6}
	KNN	3500	3725	1275	1500	1.44×10^{-5}
	Random Forest	3967	3817	1183	1033	3.36×10^{-5}
	Decision Tree	4517	3984	1016	483	8.41×10^{-6}
	Softmax Discriminant	4083	4275	725	917	1.96×10^{-4}
	Multilayer Perceptron	4050	4384	616	950	4.84×10^{-4}
	Bayesian LDC	3967	3692	1308	1033	2.50×10^{-7}

The features extracted were given to seven classifiers for performance analysis, following feature selection methods. Table 4 shows the average MSE and confusion matrix for PSO Feature Extraction with KL Divergence and IWO feature selection. The Decision Tree had the lowest MSE (9.00×10^{-6}) using PSO with KL Divergence, while the Bayesian LDC had the highest MSE (1.02×10^{-5}). With PSO and IWO, the Decision Tree had the lowest MSE (7.84×10^{-6}), while the Softmax Discriminant had the highest MSE (1.22×10^{-5}).

Table 5 displays the average MSE and confusion matrix for GWO Feature Extraction with KL Divergence and IWO feature selection methods. The results include SVM, KNN, Random Forest, Decision Tree, Softmax Discriminant, Multilayer Perceptron, and Bayesian LDC classifiers. In the GWO with KL Divergence approach, Bayesian LDC achieves the lowest MSE (1.00×10^{-8}), while the Multilayer Perceptron Classifier has the highest MSE (2.03×10^{-5}). Similarly, in the PSO with IWO approach, SVM achieves the minimum MSE (4.90×10^{-7}), while the Random Forest Classifier has the maximum MSE (1.52×10^{-5}).

Table 6 presents the mean Mean Squared Error (MSE) and confusion matrix outcomes for PSO Feature Extraction using KL Divergence and IWO feature selection techniques in Adam Hyperparameter Tuning. Among these, Bayesian LDC achieved the smallest MSE (8.41×10^{-6}) through PSO with KL Divergence, whereas Random Forest showed the highest MSE (2.72×10^{-4}). When considering PSO and IWO, Random Forest demonstrated the lowest MSE (9.00×10^{-8}), whereas Softmax Discriminant had the highest MSE (4.00×10^{-4}).

Table 4. Confusion Matrix for Classifiers for PSO with KL Divergence and IWO.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	3297	2747	2253	1703	3.24×10^{-6}
	KNN	3978	2605	2395	1022	8.41×10^{-6}
	Random Forest	4115	3294	1706	885	2.30×10^{-5}
	Decision Tree	3919	4089	911	1081	9.00×10^{-6}
	Softmax Discriminant	4089	4258	742	911	4.84×10^{-6}
	Multilayer Perceptron	4271	3633	1367	729	2.56×10^{-6}
	Bayesian LDC	3298	3311	1690	1702	1.02×10^{-5}
IWO	SVM	3854	3503	1497	1146	2.21×10^{-5}
	KNN	3490	3985	1016	1510	3.36×10^{-5}
	Random Forest	3574	2757	2243	1426	1.94×10^{-5}
	Decision Tree	2982	2871	2129	2018	7.84×10^{-6}
	Softmax Discriminant	2734	3047	1953	2266	1.22×10^{-5}
	Multilayer Perceptron	3047	2592	2408	1953	1.00×10^{-6}
	Bayesian LDC	2681	2698	2302	2319	1.85×10^{-5}

Table 5. Confusion Matrix for Classifiers for GWO with KL Divergence and IWO.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	4029	2742	2258	971	1.00×10^{-6}
	KNN	3789	4147	853	1211	4.90×10^{-5}
	Random Forest	3490	4089	911	1510	6.40×10^{-7}
	Decision Tree	3594	4147	853	1406	2.50×10^{-7}
	Softmax Discriminant	4896	2668	2333	104	1.00×10^{-6}
	Multilayer Perceptron	3737	2982	2018	1263	2.03×10^{-5}
	Bayesian LDC	3460	2767	2233	1540	1.00×10^{-8}
IWO	SVM	4401	3262	1738	599	4.90×10^{-7}
	KNN	3203	3880	1120	1797	1.60×10^{-5}
	Random Forest	4440	2735	2265	560	1.52×10^{-5}
	Decision Tree	4167	2620	2380	833	5.29×10^{-6}
	Softmax Discriminant	4219	2687	2313	781	2.30×10^{-5}
	Multilayer Perceptron	4375	2747	2253	625	9.61×10^{-6}
	Bayesian LDC	3216	2760	2240	1784	6.89×10^{-5}

Table 7 displays the average Mean Squared Error (MSE) and the results of the confusion matrices obtained from GWO Feature Extraction using KL Divergence and IWO feature selection techniques in Adam Hyperparameter Tuning. Among these approaches, Multilayer Perceptron achieved the smallest MSE of 6.40×10^{-7} using GWO with KL Divergence, while SVM exhibited the highest MSE of 1.23×10^{-5}. Considering both GWO and IWO, Decision Tree showcased the lowest MSE of 6.40×10^{-7}, whereas Softmax Discriminant had the highest MSE of 1.04×10^{-4}.

Table 6. Confusion Matrix for Classifiers: PSO with KL Divergence and IWO for Adam Hyperparameter Tuning.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	4089	3568	1433	911	6.61×10^{-4}
	KNN	4184	4487	514	817	1.44×10^{-5}
	Random Forest	4555	3520	1480	445	2.72×10^{-4}
	Decision Tree	3815	3809	1191	1185	6.72×10^{-5}
	Softmax Discriminant	4392	3948	1052	608	2.40×10^{-5}
	Multilayer Perceptron	3881	4048	952	1119	1.96×10^{-6}
	Bayesian LDC	4156	3947	1053	844	8.41×10^{-6}
IWO	SVM	3599	4085	915	1401	8.10×10^{-5}
	KNN	4058	4375	625	942	7.23×10^{-5}
	Random Forest	4129	4038	962	871	9.00×10^{-8}
	Decision Tree	3713	4308	692	1288	6.40×10^{-5}
	Softmax Discriminant	4129	4161	839	871	4.00×10^{-4}
	Multilayer Perceptron	4539	4024	976	461	2.50×10^{-5}
	Bayesian LDC	3817	3797	1203	1183	1.44×10^{-5}

Table 7. Confusion Matrix for Classifiers: GWO with KL Divergence and IWO for Adam Hyperparameter Tuning.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	3653	4466	534	1347	1.23×10^{-5}
	KNN	4139	4948	52	862	7.23×10^{-5}
	Random Forest	4044	3913	1088	956	1.30×10^{-5}
	Decision Tree	3635	3985	1016	1365	6.89×10^{-5}
	Softmax Discriminant	3565	4297	703	1435	1.37×10^{-5}
	Multilayer Perceptron	3740	4034	966	1260	6.40×10^{-7}
	Bayesian LDC	3775	3987	1013	1225	4.90×10^{-7}
IWO	SVM	4339	4617	383	661	1.94×10^{-5}
	KNN	4129	4321	680	871	5.76×10^{-6}
	Random Forest	4509	4466	534	491	7.57×10^{-5}
	Decision Tree	4617	4390	610	383	6.40×10^{-7}
	Softmax Discriminant	4409	4005	995	592	1.04×10^{-4}
	Multilayer Perceptron	4409	3913	1088	592	4.49×10^{-5}
	Bayesian LDC	4754	3973	1027	246	4.90×10^{-7}

Table 8 presents the average Mean Squared Error (MSE) and the results of confusion matrices obtained by using PSO Feature Extraction with KL Divergence and IWO feature selection techniques during R-Adam Hyperparameter Tuning. Among these methods, SVM achieved the smallest MSE of 6.56×10^{-5} when using GWO with KL Divergence, while Random Forest had the highest MSE of 1.09×10^{-5}. Considering both PSO and IWO,

Random Forest had the lowest MSE of 4.49×10^{-5}, while Softmax Discriminant had the highest MSE of 1.10×10^{-4}.

Table 8. Confusion Matrix for Classifiers: PSO with KL Divergence and IWO for RAdam Hyperparameter Tuning.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	4144	3668	1333	856	6.56×10^{-5}
	KNN	4209	4537	464	792	2.92×10^{-5}
	Random Forest	4575	3620	1380	425	1.09×10^{-5}
	Decision Tree	3950	3859	1141	1050	5.93×10^{-5}
	Softmax Discriminant	4417	4098	902	583	1.60×10^{-5}
	Multilayer Perceptron	4011	4198	802	989	3.03×10^{-5}
	Bayesian LDC	4245	4047	953	755	3.97×10^{-5}
IWO	SVM	3710	4235	765	1290	1.37×10^{-5}
	KNN	4208	4375	625	792	4.22×10^{-5}
	Random Forest	4229	4188	812	771	4.49×10^{-5}
	Decision Tree	3813	4408	592	1188	4.36×10^{-5}
	Softmax Discriminant	4229	4211	789	771	1.10×10^{-4}
	Multilayer Perceptron	4558	4074	926	443	2.30×10^{-5}
	Bayesian LDC	3917	3897	1103	1083	3.02×10^{-5}

Table 9 displays the average Mean Squared Error (MSE) and the outcomes of confusion matrices. These were derived using GWO Feature Extraction with KL Divergence and IWO feature selection methods within R-Adam Hyperparameter Tuning. Among the techniques, Bayesian LDC achieved the lowest MSE of 9.61×10^{-6} with GWO and KL Divergence. Conversely, Random Forest had the highest MSE of 1.02×10^{-5}. When considering both GWO and IWO, KNN displayed the smallest MSE of 5.48×10^{-5}, while Random Forest exhibited the highest MSE of 1.90×10^{-4}.

Table 10 presents the metrics used to evaluate the performance of classifiers, including Accuracy, Error Rate, F1 Score, MCC, Jaccard Index, g-Mean, and Kappa. The mathematical expressions for these metrics are also provided.

The lung cancer data are processed using PSO and GWO techniques to extract features from normal and malignant data. These features are then used as inputs for seven classification models. Table 11 shows the performance of the classifiers without Feature Selection. The Decision Tree Classifier stands out with the highest accuracy of 85.01% for GWO features. It also achieves the highest F1 score (85.77%), MCC value (0.70), Jaccard Index (75.08%), g-mean (85.33%), kappa score (0.70), and the lowest error rate (14.99%). In contrast, the Random Forest classifier performs poorly for PSO features, with an accuracy of 56.25%, F1 score of 55.17%, MCC value of 0.13, Jaccard Index of 38.09%, g-mean of 56.26%, kappa value of 0.13, and the highest error rate of 43.75%. Without feature selection, the Decision Tree Classifier with GWO feature extraction method achieves the best accuracy and outperforms other classifiers.

The performance of a random forest model heavily relies on the quality of individual trees and the diversity among them. If a random forest includes subpar or correlated trees, it can result in reduced overall accuracy. Correlation among trees can introduce redundant information, hampering the model's ability to generalize effectively to new data and causing a drop in accuracy.

Table 9. Confusion Matrix for Classifiers: GWO with KL Divergence and IWO for RAdam Hyperparameter Tuning.

Feature Selection	Classifiers	Confusion Matrix				MSE
		TP	TN	FP	FN	
KL Divergence	SVM	3758	4466	534	1242	1.37×10^{-5}
	KNN	4159	4948	52	842	2.40×10^{-5}
	Random Forest	4094	4063	938	906	1.02×10^{-5}
	Decision Tree	3750	3985	1016	1250	1.23×10^{-5}
	Softmax Discriminant	3670	4297	703	1330	4.76×10^{-5}
	Multilayer Perceptron	3860	4084	916	1140	2.12×10^{-5}
	Bayesian LDC	3905	4137	863	1095	9.61×10^{-6}
IWO	SVM	4439	4667	333	561	4.36×10^{-5}
	KNN	4229	4321	680	771	5.48×10^{-5}
	Random Forest	4559	4466	534	441	1.90×10^{-4}
	Decision Tree	4667	4490	510	333	2.40×10^{-5}
	Softmax Discriminant	4459	4055	945	542	5.33×10^{-5}
	Multilayer Perceptron	4459	4063	938	542	5.04×10^{-5}
	Bayesian LDC	4789	4073	927	211	1.09×10^{-5}

Table 10. Standard Benchmark Parameters.

Performance Metrics	Equation	Significance
Accuracy (%)	$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$	Average positive-to-negative sample ratio.
Error Rate	$Err = \frac{FP+FN}{TP+TN+FP+FN}$	The number of incorrect predictions, based on recorded observations.
F1 Score (%)	$F1 = \frac{2TP}{2TP+FP+FN}$	Average of precision and recall to obtain the classification accuracy of a specific class.
MCC	$MCC = \frac{TN \times TP - FN \times FP}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$	Pearson correlation between the actual output and the achieved output.
Jaccard Index (%)	$Jaccard = \frac{TP}{TP+FP+FN}$	The number of predicted true positives exceeded the number of actual positives, regardless of whether they were real or predicted.
g-mean (%)	$g - mean = \sqrt{\frac{TP}{TP+FN} * \frac{TN}{TN+FP}}$	Combination of sensitivity and specificity into a single value that balances both objectives.
Kappa	$Kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$	Inter-rater agreement measure for assessing agreement between two methods in categorizing cancer cases.

In Table 11, the Random Forest Classifier, when paired with the feature extraction technique of Particle Swarm Optimization (PSO), achieves a lower accuracy of 56.25% compared to the Genetic Wolf Optimization (GWO) approach, which achieves an accuracy of 77.84%. This discrepancy is primarily due to PSO selecting suboptimal intensity values of the segmented image, including less informative or irrelevant intensity values in the random forest model. Effective selection of intensity values is crucial for any classifier to yield better results. If the process of selecting intensity values fails to filter out irrelevant ones, it can negatively impact the performance of the Random Forest model. And also,

PSO selects intensity values that are highly specific to the training dataset, resulting in a model that performs well on the training data but struggles to generalize to new, unseen data. Furthermore, the computational demands of PSO can lead to longer tree construction times, affecting the overall classifier's performance.

Table 11. Performance Analysis of the Classifiers without Feature Selection.

Feature Extraction	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-Mean (%)	Kappa
PSO	SVM	79.53	20.47	79.40	0.59	65.83	79.53	0.59
	KNN	79.92	20.08	80.95	0.60	68.00	80.21	0.60
	Random Forest	56.25	43.75	55.17	0.13	38.09	56.26	0.13
	Decision Tree	64.01	35.99	63.89	0.28	46.94	64.01	0.28
	Softmax Discriminant	77.83	22.17	78.44	0.56	64.53	77.90	0.56
	Multilayer Perceptron	71	29	70.26	0.42	54.15	71.04	0.42
	Bayesian LDC	83.42	16.58	84.05	0.67	72.48	83.59	0.67
GWO	SVM	77.92	22.08	76.62	0.56	62.09	78.21	0.56
	KNN	72.25	27.75	71.61	0.45	55.78	72.29	0.45
	Random Forest	77.84	22.16	78.17	0.56	64.16	77.86	0.56
	Decision Tree	85.01	14.99	85.77	0.70	75.08	85.33	0.70
	Softmax Discriminant	83.58	16.42	83.26	0.67	71.32	83.62	0.67
	Multilayer Perceptron	84.34	15.66	83.80	0.69	72.12	84.46	0.69
	Bayesian LDC	76.59	23.41	77.22	0.53	62.89	76.66	0.53

Table 12 compares the performance of seven classifiers with PSO, KL Divergence, and IWO Feature Selection. The Softmax Discriminant Classifier stands out with superior results for KL Divergence features, achieving an accuracy of 83.47%, the highest F1 score of 83.18%, MCC of 0.67, Jaccard Index of 71.21%, g-mean of 83.50%, Kappa score of 0.67, and the lowest error rate of 16.53%. Conversely, the Bayesian LDC classifier performs poorly with IWO features, obtaining an accuracy of 53.79%, F1 score of 53.71%, MCC of 0.08, Jaccard Index of 36.72%, g-mean of 53.79%, Kappa of 0.08, and the highest error rate of 46.21%. Overall, the Softmax Discriminant Classifier using PSO and KL Divergence Feature selection achieves the highest accuracy and outperforms other classifiers.

Table 13 presents the performance of seven classifiers using GWO features, KL Divergence, and IWO Feature Selection. The KNN Classifier achieves the highest accuracy of 79.36% with KL Divergence features. It also obtains the highest F1 score (78.60%), MCC value (0.59), Jaccard Index (64.74%), g-mean (79.49%), kappa score (0.59), and lowest error rate (20.64%) among all classifiers. However, the Bayesian LDC classifier performs poorly with IWO features, achieving an accuracy of 59.76%, F1 score of 61.52%, MCC value of 0.20, Jaccard Index of 44.42%, g-mean of 59.84%, kappa value of 0.20, and the highest error rate (40.24%). The KNN Classifier with GWO and KL Divergence Feature selection method demonstrates the best accuracy and outperforms other classifiers.

Table 14 presents a comprehensive performance analysis of various classifiers, utilizing PSO with KL Divergence and IWO in combination with Adam Hyperparameter Tuning. The findings highlight that the K-Nearest Neighbors (KNN) Classifier attains the highest accuracy at 86.70% when incorporating KL Divergence features. This classifier also excels in other evaluation metrics, boasting the highest F1 score (86.30%), MCC value (0.74), Jaccard Index (75.96%), geometric mean (g-mean) (86.81%), kappa score (0.73), and displaying the lowest error rate (13.30%) compared to all other classifiers. Conversely, the performance of the Bayesian Linear Discriminant Classifier (LDC) is notably subpar when employing IWO features, achieving an accuracy of 76.14%, an F1 score of 76.19%, an MCC value

of 0.52, a Jaccard Index of 61.54%, a g-mean of 76.14%, a kappa value of 0.52, and the highest error rate (23.86%) among the classifiers considered. Overall, the KNN Classifier in conjunction with PSO and the KL Divergence Feature selection method emerges as the standout performer, showcasing superior accuracy and outclassing the other classifiers in the evaluation.

Table 12. Performance Analysis of the Classifiers for PSO with KL Divergence and IWO.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	60.44	39.56	62.51	0.21	45.46	60.56	0.21
	KNN	65.83	34.17	69.96	0.33	53.79	66.96	0.32
	Random Forest	74.09	25.91	76.05	0.49	61.36	74.65	0.48
	Decision Tree	80.08	19.92	79.74	0.60	66.30	80.11	0.60
	Softmax Discriminant	83.47	16.53	83.18	0.67	71.21	83.50	0.67
	Multilayer Perceptron	79.04	20.96	80.30	0.59	67.08	79.43	0.58
	Bayesian LDC	66.08	33.92	66.04	0.32	49.30	66.08	0.32
IWO	SVM	73.57	26.43	74.47	0.47	59.32	73.67	0.47
	KNN	74.74	25.26	73.43	0.50	58.01	74.95	0.49
	Random Forest	63.31	36.69	66.09	0.27	49.35	63.64	0.27
	Decision Tree	58.53	41.47	58.99	0.17	41.83	58.54	0.17
	Softmax Discriminant	57.81	42.19	56.45	0.16	39.32	57.84	0.16
	Multilayer Perceptron	56.39	43.61	58.29	0.13	41.13	56.44	0.13
	Bayesian LDC	53.79	46.21	53.71	0.08	36.72	53.79	0.08

Table 13. Performance Analysis of the Classifiers for GWO with KL Divergence and IWO.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	67.72	32.28	71.40	0.37	55.52	68.80	0.35
	KNN	79.36	20.64	78.60	0.59	64.74	79.49	0.59
	Random Forest	75.78	24.22	74.24	0.52	59.03	76.09	0.52
	Decision Tree	77.41	22.59	76.09	0.55	61.41	77.69	0.55
	Softmax Discriminant	75.64	24.37	80.08	0.57	66.77	80.74	0.51
	Multilayer Perceptron	67.19	32.81	69.49	0.35	53.25	67.54	0.34
	Bayesian LDC	62.27	37.73	64.72	0.25	47.84	62.49	0.25
IWO	SVM	76.63	23.37	79.02	0.55	65.31	77.82	0.53
	KNN	70.83	29.17	68.71	0.42	52.34	71.16	0.42
	Random Forest	71.75	28.25	75.87	0.46	61.12	74.14	0.43
	Decision Tree	67.87	32.13	72.18	0.38	56.47	69.50	0.36
	Softmax Discriminant	69.06	30.94	73.17	0.40	57.69	70.74	0.38
	Multilayer Perceptron	71.22	28.78	75.25	0.45	60.32	73.33	0.42
	Bayesian LDC	59.76	40.24	61.52	0.20	44.42	59.84	0.20

Table 14. Performance Analysis of the Classifiers: PSO with KL Divergence and IWO for Adam Hyperparameter Tuning.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	76.56	23.44	77.72	0.53	63.56	76.80	0.53
	KNN	86.70	13.30	86.30	0.74	75.96	86.81	0.73
	Random Forest	80.75	19.25	82.56	0.63	70.30	81.86	0.62
	Decision Tree	76.24	23.76	76.26	0.52	61.63	76.24	0.53
	Softmax Discriminant	83.40	16.60	84.11	0.67	72.58	83.62	0.67
	Multilayer Perceptron	79.28	20.72	78.93	0.59	65.20	79.31	0.59
	Bayesian LDC	81.03	18.97	81.42	0.62	68.67	81.08	0.62
IWO	SVM	76.84	23.16	75.66	0.54	61.84	77.05	0.54
	KNN	84.33	15.67	83.82	0.69	72.14	84.44	0.69
	Random Forest	81.67	18.33	81.83	0.63	69.25	81.68	0.63
	Decision Tree	80.21	19.79	78.95	0.61	65.23	80.56	0.60
	Softmax Discriminant	82.90	17.10	82.84	0.66	70.71	82.90	0.66
	Multilayer Perceptron	85.64	14.36	86.28	0.72	75.88	85.94	0.71
	Bayesian LDC	76.14	23.86	76.19	0.52	61.54	76.14	0.52

Table 15 provides a comprehensive analysis of classifier performance, utilizing a combination of PSO with KL Divergence and IWO along with R-Adam Hyperparameter Tuning. The results highlight that the K-Nearest Neighbors (KNN) Classifier achieves the highest accuracy of 87.45% when incorporating KL Divergence features. This classifier also excels across various evaluation metrics, including the highest F1 score (87.02%), MCC value (0.75), Jaccard Index (77.03%), geometric mean (g-mean) (87.58%), kappa score (0.75), and the lowest error rate (12.55%) compared to other classifiers. On the other hand, the Decision Tree's performance is notably weaker when using KL Divergence features, with an accuracy of 78.09%, F1 score of 78.19%, MCC value of 0.55, Jaccard Index of 64.19%, g-mean of 78.10%, kappa value of 0.54, and the highest error rate (21.91%) among considered classifiers. In summary, the KNN Classifier, in combination with PSO and the KL Divergence Feature selection method, stands out as the top performer, showcasing exceptional accuracy and surpassing other classifiers in the evaluation.

Table 16 provides a comprehensive analysis of classifier performance, utilizing GWO with KL Divergence and IWO along with Adam Hyperparameter Tuning. The results emphasize that the K-Nearest Neighbors (KNN) Classifier achieves the highest accuracy at 90.87% when incorporating KL Divergence features. This classifier also excels in various evaluation metrics, recording the highest F1 score (90.06%), MCC value (0.83), Jaccard Index (81.92%), geometric mean (g-mean) (91.71%), kappa score (0.82), and demonstrating the lowest error rate (9.14%) compared to alternative classifiers. In contrast, the performance of the Decision Tree is notably below par when utilizing KL Divergence features, attaining an accuracy of 76.20%, an F1 score of 75.33%, an MCC value of 0.53, a Jaccard Index of 60.43%, a g-mean of 76.30%, a kappa score of 0.52, and the highest error rate (23.81%) among the considered classifiers. In summary, the KNN Classifier, combined with GWO and the KL Divergence Feature selection approach, stands out as the top performer, showcasing remarkable accuracy and surpassing the other classifiers in the evaluation.

Table 15. Performance Analysis of the Classifiers: PSO with KL Divergence and IWO for RAdam Hyperparameter Tuning.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	78.11	21.89	79.11	0.56	65.44	78.32	0.56
	KNN	87.45	12.55	87.02	0.75	77.03	87.58	0.75
	Random Forest	81.95	18.05	83.53	0.65	71.71	82.92	0.64
	Decision Tree	78.09	21.91	78.19	0.55	64.19	78.10	0.54
	Softmax Discriminant	85.15	14.85	85.61	0.70	74.84	85.27	0.70
	Multilayer Perceptron	82.09	17.91	81.75	0.64	69.13	82.12	0.64
	Bayesian LDC	82.92	17.08	83.25	0.66	71.31	82.96	0.66
IWO	SVM	79.45	20.55	78.31	0.59	64.35	79.72	0.59
	KNN	85.83	14.17	85.59	0.72	74.81	85.86	0.72
	Random Forest	84.17	15.83	84.23	0.68	72.76	84.17	0.68
	Decision Tree	82.21	17.79	81.08	0.65	68.18	82.58	0.64
	Softmax Discriminant	84.40	15.60	84.43	0.69	73.05	84.40	0.69
	Multilayer Perceptron	86.32	13.68	86.95	0.73	76.91	86.59	0.73
	Bayesian LDC	78.14	21.86	78.29	0.56	64.21	78.14	0.56

Table 16. Performance Analysis of the Classifiers: GWO with KL Divergence and IWO for Adam Hyperparameter Tuning.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	81.19	18.81	79.53	0.63	66.02	81.88	0.62
	KNN	90.87	9.14	90.06	0.83	81.92	91.71	0.82
	Random Forest	79.56	20.44	79.83	0.59	66.43	79.58	0.59
	Decision Tree	76.20	23.81	75.33	0.53	60.43	76.30	0.52
	Softmax Discriminant	78.62	21.38	76.93	0.58	62.51	79.13	0.57
	Multilayer Perceptron	77.74	22.26	77.07	0.56	62.69	77.82	0.55
	Bayesian LDC	77.62	22.38	77.14	0.55	62.78	77.66	0.55
IWO	SVM	89.56	10.44	89.27	0.79	80.61	89.66	0.79
	KNN	84.50	15.51	84.19	0.69	72.70	84.54	0.69
	Random Forest	89.76	10.24	89.80	0.80	81.49	89.76	0.80
	Decision Tree	90.07	9.93	90.29	0.80	81.30	90.13	0.80
	Softmax Discriminant	84.13	15.87	84.75	0.68	73.54	84.31	0.68
	Multilayer Perceptron	83.21	16.79	84.00	0.67	72.42	83.47	0.66
	Bayesian LDC	87.27	12.73	88.19	0.75	78.88	88.00	0.75

Table 17 presents a comprehensive analysis of classifier performance, utilizing GWO with KL Divergence, and IWO alongside R-Adam Hyperparameter Tuning. The outcomes underscore the Decision Tree Classifier's exceptional performance, achieving the highest accuracy at 91.57% when integrating KL Divergence features. This classifier also outperforms others across various evaluation metrics, achieving the highest F1 score (91.71%), MCC value (0.83), Jaccard Index (84.70%), geometric mean (g-mean) (91.87%), kappa score (0.83), and demonstrating the lowest error rate (8.43%) compared to alternative classifiers. In

contrast, the performance of the Decision Tree notably drops when utilizing IWO, with an accuracy of 77.35%, an F1 score of 76.80%, an MCC value of 0.55, a Jaccard Index of 62.34%, a g-mean of 77.39%, a kappa score of 0.55, and the highest error rate (22.66%) among the considered classifiers. In recap, the Decision Tree classifier, combined with GWO and the KL Divergence Feature selection approach, emerges as the leading performer, showcasing remarkable accuracy and surpassing the other classifiers in the evaluation.

Table 17. Performance Analysis of the Classifiers: GWO with KL Divergence and IWO for RAdam Hyperparameter Tuning.

Feature Selection	Classifiers	Accuracy (%)	Error Rate (%)	F1 Score (%)	MCC	Jaccard Index (%)	g-mean (%)	Kappa
KL Divergence	SVM	82.24	17.76	80.89	0.65	67.91	82.77	0.64
	KNN	91.07	8.94	90.30	0.82	82.31	91.61	0.82
	Random Forest	81.56	18.44	81.62	0.63	68.95	81.56	0.63
	Decision Tree	77.35	22.66	76.80	0.55	62.34	77.39	0.55
	Softmax Discriminant	79.67	20.33	78.31	0.60	64.35	80.05	0.59
	Multilayer Perceptron	79.44	20.56	78.97	0.59	65.25	79.49	0.59
	Bayesian LDC	80.42	19.58	79.96	0.61	66.61	80.47	0.61
IWO	SVM	91.06	8.94	90.86	0.82	83.24	91.13	0.82
	KNN	85.50	14.51	85.36	0.71	74.46	85.50	0.71
	Random Forest	90.26	9.74	90.35	0.81	82.39	90.27	0.81
	Decision Tree	91.57	8.43	91.71	0.83	84.70	91.87	0.83
	Softmax Discriminant	85.13	14.87	85.71	0.70	75.00	85.32	0.70
	Multilayer Perceptron	85.21	14.79	85.77	0.71	75.09	85.39	0.70
	Bayesian LDC	88.62	11.38	89.38	0.78	80.80	89.25	0.77

Table 18 presents a summary of the performance outcomes for each combination of feature extraction and feature selection using Adam and R-Adam Hyperparameter tuning methods across all seven classifiers. The highest accuracy of 91.57% in the Decision Tree classifier was attained by combining GWO and IWO techniques, utilizing the RAdam Hyperparameter tuning approach.

Figure 7 illustrates the comparative performance of classifiers in relation to Accuracy, both with and without the integration of feature selection. As depicted in the graph, among all of the classifier types, the Decision Tree classifier employing the GWO (Grey Wolf Optimization) feature extraction method outperformed the rest in terms of achieving the highest accuracy. When utilizing the KL Divergence feature selection technique, along with PSO feature extraction technique, the Softmax Discriminant Classifier demonstrated a commendable accuracy of 83.47%. Similarly, when employing the IWO (Invasive Weed Optimization) feature selection technique, along with GWO feature extraction technique, the SVM (Support Vector Machine) classifier exhibited a notable accuracy of 76.63%. In contrast, the Mathematical feature selection approaches yielded comparatively lower accuracy when compared to scenarios where feature selection was not applied.

Figure 8 displays how classifiers perform when hyperparameter tuning methods such as Adam and RAdam are used to enhance accuracy. Even after applying feature selection techniques, there's no significant improvement in classifier accuracy compared to using no feature selection. To address this, hyperparameter update algorithms are introduced. The accuracy achieved through the KL Divergence feature selection method is notably high across all classifiers. However, for the IWO feature selection technique, accuracy seems to be somewhat lower. This prompts the use of hyperparameter update algorithms specifically for the IWO feature selection. As a result of employing these algorithms, there's

a substantial accuracy improvement for all classifiers using the IWO feature selection. Notably, the Decision Tree classifier combined with GWO feature extraction and IWO feature selection, along with the RAdam hyperparameter update algorithm, achieves the highest accuracy at 91.57%.

Table 18. Performance Analysis of the classifiers for Maximum Accuracy.

S No	Feature Extraction	Feature Selection	Classifiers	Accuracy (%)
1	PSO	-	Bayesian LDC	83.42%
2	GWO	-	Decision Tree	85.01%
3	PSO	KL Divergence	Softmax Discriminant	83.47%
4	PSO	IWO	KNN	74.74%
5	GWO	KL Divergence	KNN	79.36%
6	GWO	IWO	SVM	76.63%
7	PSO	KL Divergence	KNN with Adam	86.70%
8	PSO	IWO	MLP with Adam	85.64%
9	PSO	KL Divergence	KNN with RAdam	87.45%
10	PSO	IWO	MLP with RAdam	86.32%
11	GWO	KL Divergence	KNN with Adam	90.87%
12	GWO	IWO	Decision Tree with Adam	90.07%
13	GWO	KL Divergence	KNN with RAdam	91.07%
14	GWO	IWO	Decision Tree with RAdam	91.57%
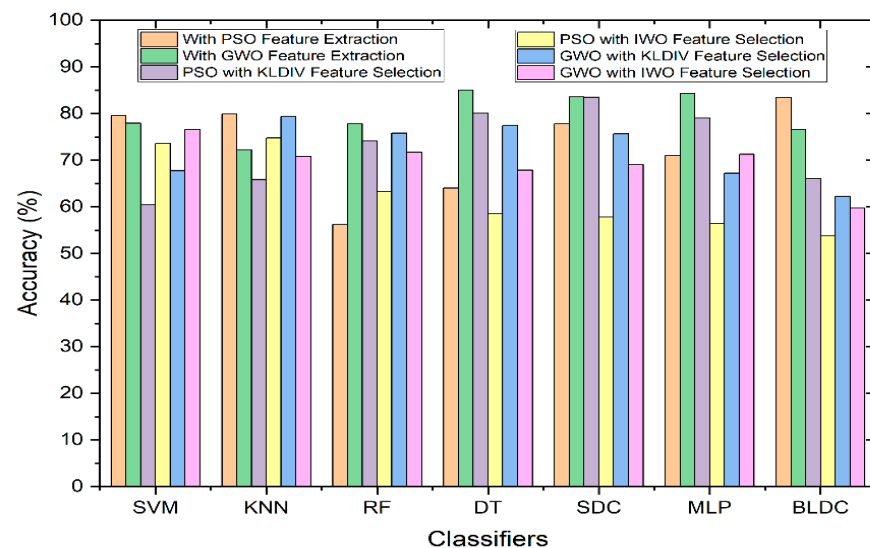
Figure 7. Performance of Classifiers with and without Feature Selection Methods in terms of Accuracy.

Figure 9 illustrates the classifiers' performance by analyzing the Deviation of MCC and Kappa parameters in relation to their mean values. These parameters, MCC and Kappa, serve as benchmarks for evaluating how classifiers respond to different inputs. The study involves two input categories: features extracted using PSO and GWO, followed by feature selection through KL Divergence and IWO. The selected features are then inputted into the classifiers, and their effectiveness is evaluated through the resulting MCC and Kappa values. The average MCC and Kappa values attained from the classifiers are 0.56661 and 0.56256, respectively. A methodology is devised to assess classifier performance by examining the variability of MCC and Kappa values from their respective means. Notably,

Figure 9 depicts a trend where MCC and Kappa values in the graph's third quadrant correspond to non-linear outcomes with lower performance metrics. Conversely, values in the graph's first quadrant indicate improved classifier performance, with MCC and Kappa values surpassing the average. This pattern suggests an enhancement in classifier performance for GWO inputs when coupled with IWO feature selection, particularly within the context of the RAdam hyperparameter tuning approach. Figure 9 is also characterized by a linear curve fitting described by the equation $Y = 1.0017\,X + 4 \times 10^{-06}$, with an R^2 value of 0.998.
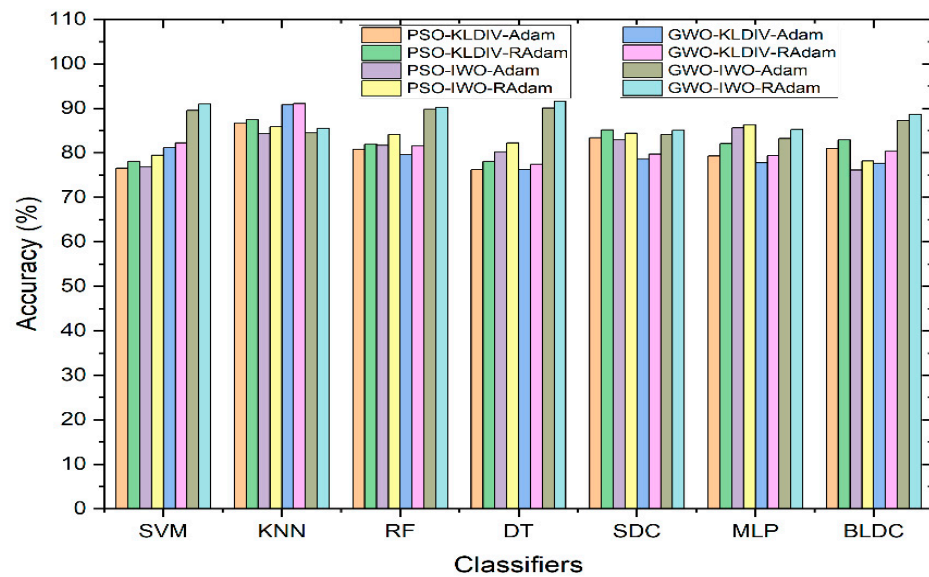
Figure 8. Performance of Classifiers with Hyperparameter Tuning Methods for Adam and RAdam in terms of Accuracy.
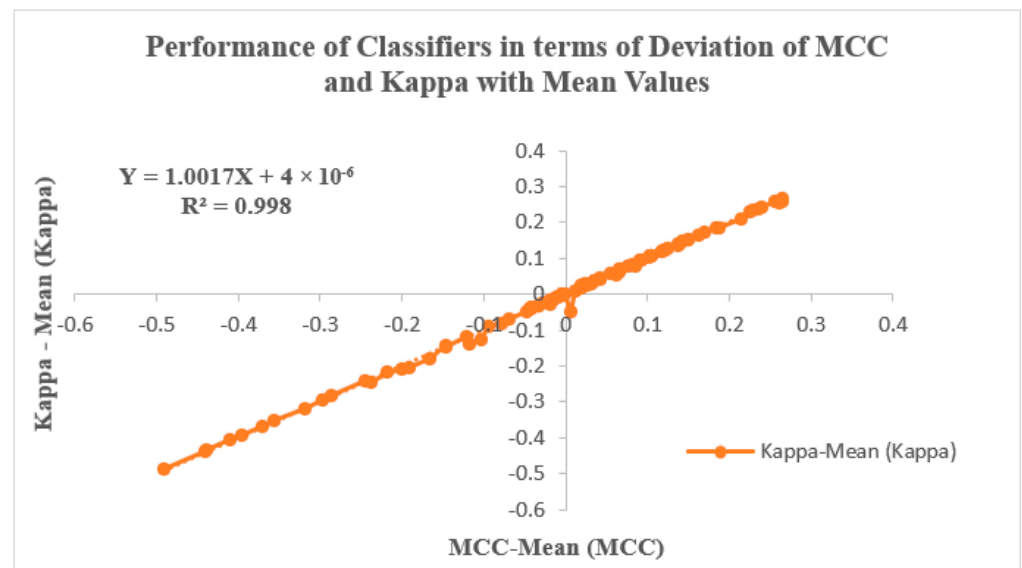
Figure 9. Performance of Classifiers in terms of Deviation of MCC and Kappa Parameters with mean values.

6.4. Computational Complexity Analysis of the Classifiers

Computational complexity also acts as a performance metric for classifiers, encompassing time and space complexities. This study utilizes the Big O notation to characterize the computational complexity of feature extraction, feature selection, and classification

methods. The assessment of Computational Complexity involves an input size labelled as '*n*'. When the input size is $O(1)$, the computational complexity remains minimal. However, as the input size increases, so does the computational complexity. The relationship between input size and computational complexity is encapsulated by the Big O notation. Specifically, if the complexity grows logarithmically with the increase in 'n', it is represented as $O(\log n)$. The classifiers examined in this research integrate either feature extraction methods, feature selection techniques, or a combination of both. Hence, computational complexity becomes a blend of these hybrid methodologies. Table 19 offers an overview of the computational complexity associated with the classifiers across diverse Feature Extraction and Feature Selection Techniques.

Table 19. Computational Complexity of the classifiers among Feature Extraction, Feature Selection and Hyperparameter Tuning approaches.

S No	Classifiers	Without Feature Extraction	With Feature Extraction		With Feature Selection		With Hyperparameter Tuning of IWO Feature Selection Method	
			PSO	GWO	KL Divergence	IWO	Adam	RAdam
1	SVM	$O\left(2n^2\right)$	$O\left(2n^5\right)$	$O\left(2n^5\right)$	$O\left(2n^6\right)$	$O\left(2n^6\log n\right)$	$O\left(2n^2\log n\right)$	$O\left(4n^7\log 5n\right)$
2	KNN	$O\left(n^2\right)$	$O\left(n^5\right)$	$O\left(n^5\right)$	$O\left(n^6\right)$	$O\left(n^6\log n\right)$	$O\left(2n^7\log 2n\right)$	$O\left(2n^7\log 5n\right)$
3	RF	$O\left(n\log n\right)$	$O\left(n^4\log n\right)$	$O\left(n^4\log n\right)$	$O\left(n^5\log n\right)$	$O\left(n^5\log 2n\right)$	$O\left(2n^6\log 3n\right)$	$O\left(2n^6\log 6n\right)$
4	DT	$O\left(\log n\right)$	$O\left(n^3\log n\right)$	$O\left(n^3\log n\right)$	$O\left(n^4\log n\right)$	$O\left(n^4\log 2n\right)$	$O\left(2n^5\log 3n\right)$	$O\left(2n^5\log 6n\right)$
5	SDC	$O\left(n^2\right)$	$O\left(n^5\right)$	$O\left(n^5\right)$	$O\left(n^6\right)$	$O\left(n^6\log n\right)$	$O\left(2n^7\log 2n\right)$	$O\left(2n^7\log 5n\right)$
6	MLP	$O\left(n^5\right)$	$O\left(n^8\right)$	$O\left(n^8\right)$	$O\left(n^9\right)$	$O\left(n^9\log n\right)$	$O\left(2n^{10}\log 2n\right)$	$O\left(2n^{10}\log 5n\right)$
7	BLDC	$O\left(n^2\right)$	$O\left(n^5\right)$	$O\left(n^5\right)$	$O\left(n^6\right)$	$O\left(n^6\log n\right)$	$O\left(2n^7\log 2n\right)$	$O\left(2n^7\log 5n\right)$

As evident from Table 19, when feature extraction techniques are not employed, classifiers such as SVM, KNN, Random Forest (RF), Decision Tree (DT), Softmax Discriminant classifier (SDC), Multilayer Perceptron (MLP), and Bayesian LDC (BLDC) exhibit lower levels of computational complexity. When utilizing the GWO feature extraction technique, the Decision Tree classifier stands out with a computational complexity of $O\left(n^3\log n\right)$ and achieves a high accuracy of 85.01%. However, the BLDC classifier, with a computational complexity of $O\left(n^6\log n\right)$ for GWO feature extraction, performs poorly when IWO feature selection methods are applied across the classifiers. The observed underperformance is linked to outlier problems present in the GWO features. To improve classifier performance, this study integrates hyperparameter tuning into the IWO feature selection method. Particularly, the Decision Tree classifier demonstrates remarkable performance with accuracies of 90.07% and 91.57% when utilizing GWO feature extraction in conjunction with IWO. These improvements come with a moderate computational complexity represented by $O\left(2n^5\log 3n\right)$ for the Adam Hyperparameter tuning approach and $O\left(2n^5\log 6n\right)$ for the RAdam Hyperparameter tuning approach.

6.5. Comparison of Previous Works

Comparison charts with different Machine Learning and Deep Learning models along with classifiers are shown in the Table 20 for the different datasets of lung cancer. As noted in the Table 20, that the classifier performance is analyzed for the four different datasets, namely CRAG, LIDC-IDRI, LUNA16 and LC25000. The following classifiers namely Ensemble, ResNet50, KNN, AlexNet and CNN were analyzed. As shown in the Table 20., for the CRAG database, the ResNet50 attains the maximum accuracy of 93.91% whereas for CT image database (LIDC-IDRI, LUNA16), the CNN-ALCDC model attains the maximum accuracy of 97.2%. This is due to the smaller number of CT images as the

data set. Similarly, for LC25000 database, for a binary classification problem, 5000 images for each class are taken, which attains a commendable accuracy of 91.57%.

Table 20. Comparison of classifier performance with different datasets.

S No	Authors	Dataset Used	Machine Learning Models/Classifiers	Accuracy (%)
1	Bukhari, S. et al. [57]	CRAG Dataset	ResNet-50	93.91%
2	Jinsa Kuruvilla. et al. [58]	LIDC Dataset (155 Patients—CT images)	Feed Forward Back Propagation Neural Networks	93.3%
3	Dabass, M. et al. [59]	CRAG Dataset	Atrous Convolved Hybrid Seg-Net Architecture	87.63%
4	Supriya, Suresh.et al. [60]	LIDC-IDRI Repository (CT scans)	CNN	93.9%
5	Wadood, Abdul. [61]	LIDC-IDRI Repository (CT scans)	CNN-ALCDC	97.2%
6	Rekka, Mastouri. et al. [62]	LUNA16 Database (3186 CT images)	BCNN [VGG16, VGG19]	91.99%
7	Tasnim, Ahmed. et al. [63]	LUNA16 Database	3D CNN	80%
8	Mesut Toğaçar. et al. [64]	Cancer Imaging Archieve (CT images)	AlexNet and kNN classifier	98.74%
9	Anum, Masood. et al. [65]	Biomedical Datasets–IoT	CNN DFCNet	77.6% 84.58%
10	Wahyudi, Setiawan. et al. [66]	LC25000 Database	CNN	87.16%
11	Manaswini, Pradhan. [67]	LC25000 Database	Without Feature Selection (EGOA)—KNN With Feature Selection (EGOA)–KNN	80.16% 81.59%
12	Phankokkruad, M [68]	LC25000 Database	Ensemble ResNet50V2	91% 90%
13	Karthikeyan Shanmugam, Harikumar Rajaguru	LC25000 Database	Feature Extraction-GWO Feature Selection—IWO Decision tree with RAdam Hyper parameter Updation method	91.57%

7. Conclusions

Early diagnosis of lung cancer enhances patient life expectancy. This paper proposes machine learning techniques to enhance classifier accuracy and enable early identification using histopathological images. The primary aim is to achieve lung cancer classification with high accuracy, while minimizing false positives and false negatives. The study applies adaptive median filtering and a modified KFCM-based segmentation method to obtain the segmented images for better classification results. Feature extraction involves optimization techniques such as PSO and GWO which reduces the dimensionality of the segmented image to [512 × 10], followed by statistical analysis. Feature selection reduces the number of intensity values to [100 × 12] for lung cancer classification. Through the utilization of KL Divergence and Invasive Weed Optimization to evaluate the dimensionally reduced features, the datasets undergo classification with various classifiers to achieve better accuracy. The classification process entails seven classifiers, coupled with Hyperparameter selection using Adam and Radam methods, which are compared and analyzed. The Decision Tree Classifier for GWO features without feature selection achieves a better accuracy of 85.01%.

Mathematical feature selection methods had lower accuracy compared to scenarios without feature selection. The results are further enhanced when the Hyperparameter Up-to-date methods are employed which reveal that the combination of GWO-IWO-Decision Tree classifier for RAdam outperforms all other classifiers, achieving an overall accuracy of 91.57% in classifying Benign and Adenocarcinoma classes. Future research directions will explore diverse feature selection techniques, optimization methodologies, and the inclusion of deep learning approaches such as CNN, DNN, and LSTM to further enhance lung cancer classification.

Author Contributions: Conceptualization, K.S.; Methodology, H.R.; Software, K.S.; Validation, H.R.; Formal analysis, K.S.; Investigation, H.R.; Resources, K.S.; Data curation, K.S.; Writing-original draft, K.S.; Writing-review and editing, H.R.; Visualization, H.R.; Supervision, H.R. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: Not Applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Prabhakar, S.K.; Lee, S.W. An Integrated Approach for Ovarian Cancer Classification with the Application of Stochastic Optimization. *IEEE Access.* **2020**, *8*, 127866–127882. [CrossRef]
2. Jemal, A.; Ward, E.M.; Johnson, C.J.; Cronin, K.A.; Ma, J.; Ryerson, A.B.; Mariotto, A.; Lake, A.J.; Wilson, R.; Sherman, R.L.; et al. Annual Report to the Nation on the Status of Cancer, 1975–2014, Featuring Survival. *J. Natl. Cancer Inst.* **2017**, *109*, djx030. [CrossRef] [PubMed]
3. Miki, T.; Yano, S.; Hanibuchi, M.; Sone, S. Bone Metastasis Model with Multiorgan Dissemination of Human Small-Cell Lung Cancer (SBC-5) Cells in Natural Killer Cell-Depleted SCID Mice. *Oncol. Res. Featur. Preclin. Clin. Cancer Ther.* **2001**, *12*, 209–217. [CrossRef] [PubMed]
4. Mahmood, H.; Shaban, M.; Indave, B.I.; Santos-Silva, A.R.; Rajpoot, N.; Khurram, S.A. Use of artificial intelligence in diagnosis of head and neck precancerous and cancerous lesions: A systematic review. *Oral Oncol.* **2020**, *110*, 104885. [CrossRef] [PubMed]
5. He, L.; Long, L.R.; Antani, S.; Thoma, G.R. Histology image analysis for carcinoma detection and grading. *Comput. Methods Programs Biomed.* **2012**, *107*, 538–556. [CrossRef] [PubMed]
6. Komura, D.; Ishikawa, S. Machine Learning Methods for Histopathological Image Analysis. *Comput. Struct. Biotechnol. J.* **2018**, *16*, 34–42. [CrossRef]
7. Andreadis, D.A.; Pavlou, A.M.; Panta, P. Biopsy and oral squamous cell carcinoma histopathology. In *Oral Cancer Detection: Novel Strategies and Clinical Impact*; Springer International Publishing: Cham, Switzerland, 2019; pp. 133–151.
8. Gertych, A.; Swiderska-Chadaj, Z.; Ma, Z.; Ing, N.; Markiewicz, T.; Cierniak, S.; Salemi, H.; Guzman, S.; Walts, A.E.; Knudsen, B.S. Convolutional neural networks can accurately distinguish four histologic growth patterns of lung adenocarcinoma in digital slides. *Sci. Rep.* **2019**, *9*, 1483. [CrossRef]
9. Zhang, X.; Xing, F.; Su, H.; Yang, L.; Zhang, S. High-throughput histopathological image analysis via robust cell segmentation and hashing. *Med. Image Anal.* **2015**, *26*, 306–315. [CrossRef]
10. Draelos, R.L.; Dov, D.; Mazurowski, M.A.; Lo, J.Y.; Henao, R.; Rubin, G.D.; Carin, L. Machine-learning-based multiple abnormality prediction with large-scale chest computed tomography volumes. *Med. Image Anal.* **2021**, *67*, 101857. [CrossRef]
11. Szabó, M.; Bozó, A.; Darvas, K.; Soós, S.; Őzse, M.; Iványi, Z.D. The role of ultrasonographic lung aeration score in the prediction of postoperative pulmonary complications: An observational study. *BMC Anesthesiol.* **2021**, *21*, 19. [CrossRef]
12. Toğaçar, M. Disease type detection in lung and colon cancer images using the complement approach of inefficient sets. *Comput. Biol. Med.* **2021**, *137*, 104827. [CrossRef] [PubMed]
13. Talukder, M.A.; Islam, M.M.; Uddin, M.A.; Akhter, A.; Hasan, K.F.; Moni, M.A. Machine Learning-based Lung and Colon Cancer Detection using Deep Feature Extraction and Ensemble Learning. *Expert Syst. Appl.* **2022**, *205*, 117695. [CrossRef]
14. Ozekes, S.; Camurcu, A.Y. Automatic lung nodule detection using template matching. In Proceedings of the 4th International Conference on Advances in Information Systems, ADVIS, Izmir, Turkey, 18–20 October 2006; Volume 4243, pp. 247–253.
15. Schilham, A.M.; van Ginneken, B.; Loog, M. A computer-aided diagnosis system for detection of lung nodules in chest radiographs with an evaluation on a public database. *Med. Image Anal.* **2006**, *10*, 247–258. [CrossRef]
16. Wang, S.; Wang, T.; Yang, L.; Yang, D.M.; Fujimoto, J.; Yi, F.; Luo, X.; Yang, Y.; Yao, B.; Lin, S.; et al. ConvPath: A software tool for lung adenocarcinoma digital pathological image analysis aided by a convolutional neural network. *eBioMedicine* **2019**, *50*, 103–110. [CrossRef]

17. Dehmeshki, J.; Ye, X.; Lin, X.; Valdivieso, M.; Amin, H. Automated detection of lung nodules in CT images using shape-based genetic algorithm. *Comput. Med. Imaging Graph.* **2007**, *31*, 408–417. [CrossRef] [PubMed]

18. Suárez-Cuenca, J.J.; Tahoces, P.G.; Souto, M.; Lado, M.J.; Remy-Jardin, M.; Remy, J.; Vidal, J.J. Application of the iris filter for automatic detection of pulmonary nodules on computed tomography images. *Comput. Biol. Med.* **2009**, *39*, 921–933. [CrossRef] [PubMed]

19. Murphy, K.; van Ginneken, B.; Schilham, A.M.; de Hoop, B.J.; Gietema, H.A.; Prokop, M. A large-scale evaluation of automatic pulmonary nodule detection in chest CT using local image features and k-nearest-neighbour classification. *Med. Image Anal.* **2009**, *13*, 757–770. [CrossRef]

20. Wei, J.W.; Tafe, L.J.; Linnik, Y.A.; Vaickus, L.J.; Tomita, N.; Hassanpour, S. Pathologist-level classification of histologic patterns on resected lung adenocarcinoma slides with deep neural networks. *Sci. Rep.* **2019**, *9*, 3358. [CrossRef]

21. Al-Jabbar, M.; Alshahrani, M.; Senan, E.M.; Ahmed, I.A. Histopathological Analysis for Detecting Lung and Colon Cancer Malignancies Using Hybrid Systems with Fused Features. *Bioengineering* **2023**, *10*, 383. [CrossRef]

22. Teramoto, A.; Tsukamoto, T.; Kiriyama, Y.; Fujita, H. Automated Classification of Lung Cancer Types from Cytological Images Using Deep Convolutional Neural Networks. *Biomed. Res. Int.* **2017**, *2017*, 4067832. [CrossRef]

23. Shapcott, M.; Hewitt, K.J.; Rajpoot, N. Deep Learning with Sampling in Colon Cancer Histology. *Front. Bioeng. Biotechnol.* **2019**, *7*, 52. [CrossRef] [PubMed]

24. Barker, J.; Hoogi, A.; Depeursinge, A.; Rubin, D.L. Automated classification of brain tumor type in whole-slide digital pathology images using local representative tiles. *Med. Image Anal.* **2016**, *30*, 60–71. [CrossRef] [PubMed]

25. Ojansivu, V.; Linder, N.; Rahtu, E.; Pietikäinen, M.; Lundin, M.; Joensuu, H.; Lundin, J. Automated classification of breast cancer morphology in histopathological images. *Diagn. Pathol.* **2013**, *8*, S29. [CrossRef]

26. Ficsor, L.; Varga, V.S.; Tagscherer, A.; Tulassay, Z.; Molnar, B. Automated classification of inflammation in colon histological sections based on digital microscopy and advanced image analysis. *Cytom. Part A* **2008**, *73*, 230–237. [CrossRef]

27. Taheri, M.; Hamer, G.; Son, S.H.; Shin, S.Y. Automated Single and Multi-Breast Tumor Segmentation Using Improved Watershed Technique in 2D MRI Images. In Proceedings of the International Conference on Research in Adaptive and Convergent Systems (RACS '16), Association for Computing Machinery, New York, NY, USA, 11–14 October 2016; pp. 61–66.

28. Huang, P.W.; Lai, Y.H. Effective segmentation and classification for HCC biopsy images. *Pattern Recognit.* **2010**, *43*, 1550–1563. [CrossRef]

29. Gessert, N.; Bengs, M.; Wittig, L.; Drömann, D.; Keck, T.; Schlaefer, A.; Ellebrecht, D.B. Deep transfer learning methods for colon cancer classification in confocal laser microscopy images. *Int. J. Comput. Assist. Radiol. Surg.* **2019**, *14*, 1837–1845. [CrossRef]

30. Sinha, N.; Ramakrishnan, A.G. Automation of differential blood count. In Proceedings of the TENCON 2003, Conference on Convergent Technologies for Asia-Pacific Region, Bangalore, India, 15–17 October 2003; Volume 2, pp. 547–551.

31. Fauziah, K.; Anton, S.P.; Abdullah, A. Detection of leukemia in human blood sample based on microscopic images: A study. *J. Theor. Appl. Inf. Technol.* **2012**, *46*, 579–586.

32. Chen, C.H.; Lee, Y.W.; Huang, Y.S.; Lan, W.R.; Chang, R.F.; Tu, C.Y.; Chen, C.Y.; Liao, W.C. Computer-aided diagnosis of endobronchial ultrasound images using convolutional neural network. *Comput. Methods Programs Biomed.* **2019**, *177*, 175–182. [CrossRef]

33. Khoirunnisa, A.; Adytia, D. Implementation of CRNN Method for Lung Cancer Detection based on Microarray Data. *Int. J. Inform. Vis.* **2023**, *7*, 2. [CrossRef]

34. Mehmood, S.; Ghazal, T.M.; Khan, M.A.; Zubair, M.; Naseem, M.T.; Faiz, T.; Ahmad, M. Malignancy Detection in Lung and Colon Histopathology Images Using Transfer Learning with Class Selective Image Processing. *IEEE Access* **2022**, *10*, 25657–25668. [CrossRef]

35. Boumaraf, S.; Liu, X.; Zheng, Z.; Ma, X.; Ferkous, C. A new transfer learning based approach to magnification dependent and independent classification of breast cancer in histopathological images. *Biomed. Signal Process. Control.* **2021**, *63*, 102192. [CrossRef]

36. Rodríguez Fernández, A.; Gómez Río, M.; Llamas Elvira, J.M.; Sánchez-Palencia Ramos, A.; Bellón Guardia, M.; Ramos Font, C.; Torné Poyatos, P.; Pedraza Muriel, V. Diagnosis efficacy of structural (CT) and functional (FDG-PET) imaging methods in the thoracic and extrathoracic staging of non-small cell lung cancer. *Clin. Transl. Oncol.* **2007**, *9*, 32–39. [CrossRef] [PubMed]

37. Spanhol, F.A.; Oliveira, L.; Cavalin, P.R.; Petitjean, C.; Heutte, L. Deep features for breast cancer histopathological image classification. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 1868–1873.

38. Ma, Z.; Tavares, J.M.R.; Jorge, R.N. A review on the current segmentation algorithms for medical images. In Proceedings of the 1st International Conference on Imaging Theory and Applications (IMAGAPP), Lisboa, Portugal, 5–8 February 2009.

39. Ding, Y.; Fu, X. Kernel-based fuzzy c-means clustering algorithm based on genetic algorithm. *Neurocomputing* **2016**, *188*, 233–238. [CrossRef]

40. Ciompi, F.; Geessink, O.G.; Bejnordi, B.E.; Souza, G.S.; Baidoshvili, A.; Litjens, G.J.; Ginneken, B.V.; Nagtegaal, I.D.; Laak, J.V. The importance of stain normalization in colorectal tissue classification with convolutional networks. In Proceedings of the 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), Melbourne, Australia, 18–21 April 2017; pp. 160–163.

41. Liu, X.; Wang, C.; Bai, J.; Liao, G. Fine-tuning Pre-trained Convolutional Neural Networks for Gastric Precancerous Disease Classification on Magnification Narrow-band Imaging Images. *Neurocomputing* **2020**, *392*, 253–267. [CrossRef]

42. Sun, C.; Xu, A.; Liu, D.; Xiong, Z.; Zhao, F.; Ding, W. Deep Learning-Based Classification of Liver Cancer Histopathology Images Using Only Global Labels. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 1643–1651. [CrossRef]

43. Alinsaif, S.; Lang, J. Texture features in the Shearlet domain for histopathological image classification. *BMC Med. Inform. Decis. Mak.* **2020**, *20* (Suppl. S14), 312. [CrossRef]

44. Sujin, P.R.; Prakash, T.R.; Linda, M.M. Particle Swarm Optimization Based Reactive Power Optimization. *ArXiv* **2010**, arXiv:1001.3491.

45. Saremi, S.; Mirjalili, S.Z.; Mirjalili, S.M. Evolutionary population dynamics and grey wolf optimizer. *Neural Comput. Appl.* **2015**, *26*, 1257–1263. [CrossRef]

46. Prabhakar, S.K.; Rajaguru, H. Alcoholic EEG signal classification with Correlation Dimension based distance metrics approach and Modified Adaboost classification. *Heliyon* **2020**, *6*, e05689. [CrossRef]

47. Mallahzadeh, A.R.; Oraizi, H.; Davoodi-Rad, Z. Application of The Invasive Weed Optimization Technique for Antenna Configurations. *Prog. Electromagn. Res.* **2008**, *79*, 425–428. [CrossRef]

48. Xia, L.; Huang, Q.; Wu, D. Decision Tree-Based Contextual Location Prediction from Mobile Device Logs. *Mob. Inf. Syst.* **2018**, *2018*, 1852861. [CrossRef]

49. Kumar, M.; Rath, N.K.; Swain, A.; Rath, S.K. Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor. *Procedia Comput. Sci.* **2015**, *54*, 301–310. [CrossRef]

50. Gibson, R.; Danaher, T.; Hehir, W.; Collins, L. A remote sensing approach to mapping fire severity in south-eastern Australia using sentinel 2 and random forest. *Remote Sens. Environ.* **2020**, *240*, 111702–111715. [CrossRef]

51. Zang, F.; Zhang, J. Softmax Discriminant Classifier. In Proceedings of the Third International Conference on Multimedia Information Networking and Security, Shanghai, China, 4–6 November 2011; pp. 16–19.

52. Smith, K.A.; Gupta, J.N.D. Neural networks in business: Techniques and applications for the operations researcher. *Comput. Oper. Res.* **2000**, *27*, 1023–1044. [CrossRef]

53. Cybenko, G.V. Approximation by superpositions of a sigmoidal function. *Math. Control. Signals Syst.* **1989**, *2*, 303–314. [CrossRef]

54. Fonseca, P.; Den Teuling, N.; Long, X.; Aarts, R.M. Cardiorespiratory Sleep Stage Detection Using Conditional Random Fields. *IEEE J. Biomed. Health Inform.* **2017**, *21*, 956–966. [CrossRef]

55. Claesen, M.; Moor, B.D. Hyperparameter Search in Machine Learning. *arXiv* **2015**, arXiv:1502.02127.

56. Sen, S.Y.; Ozkurt, N. Convolutional Neural Network Hyperparameter Tuning with Adam Optimizer for ECG Classification. In Proceedings of the 2020 Innovations in Intelligent Systems and Applications Conference, ASYU, Istanbul, Turkey, 15–17 October 2020; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2020.

57. Bukhari, S.U.K.; Syed, A.; Bokhari, S.K.A.; Hussain, S.S.; Armaghan, S.U.; Shah, S.S.H. The histological diagnosis of colonic adenocarcinoma by applying partial self supervised learning. *medRxiv* **2020**. [CrossRef]

58. Kuruvilla, J.; Gunavathi, K. Lung cancer classification using neural networks for CT images. *Comput. Methods Programs Biomed.* **2014**, *113*, 202–209. [CrossRef]

59. Dabass, M.; Dabass, J. An Atrous Convolved Hybrid Seg-Net Model with residual and attention mechanism for gland detection and segmentation in histopathological images. *Comput. Biol. Med.* **2023**, *155*, 106690. [CrossRef]

60. Suresh, S.; Mohan, S. ROI-based feature learning for efficient true positive prediction using convolutional neural network for lung cancer diagnosis. *Neural Comput. Appl.* **2020**, *32*, 15989–16009. [CrossRef]

61. Abdul, W. An Automatic Lung Cancer Detection and Classification (ALCDC) System Using Convolutional Neural Network. In Proceedings of the 2020 13th International Conference on Developments in eSystems Engineering (DeSE), Liverpool, UK, 14–17 December 2020; pp. 443–446.

62. Mastouri, R.; Khlifa, N.; Neji, H.; Hantous-Zannad, S. A bilinear convolutional neural network for lung nodules classification on CT images. *Int. J. Comput. Assist. Radiol. Surg.* **2021**, *16*, 91–101. [CrossRef] [PubMed]

63. Ahmed, T.; Parvin, M.S.; Haque, M.R.; Uddin, M.S. Lung Cancer Detection Using CT Image Based on 3D Convolutional Neural Network. *J. Comput. Chem.* **2020**, *8*, 35–42. [CrossRef]

64. Toğaçar, M.; Ergen, B.; Cömert, Z. Detection of lung cancer on chest CT images using minimum redundancy maximum relevance feature selection method with convolutional neural networks. *Biocybern. Biomed. Eng.* **2020**, *40*, 23–39. [CrossRef]

65. Masood, A.; Sheng, B.; Li, P.; Hou, X.; Wei, X.; Qin, J.; Feng, D. Computer-Assisted Decision Support System in Pulmonary Cancer detection and stage classification on CT images. *J. Biomed. Inform.* **2018**, *79*, 117–128. [CrossRef]

66. Setiawan, W.; Suhadi, M.M.; Husni; Pramudita, Y.D. Histopathology of Lung Cancer Classification Using Convolutional Neural Network with Gamma Correction. Commun. *Math. Biol. Neurosci.* **2022**, *2022*, 81.

67. Pradhan, M.; Bhuiyan, A.; Mishra, S.; Thieu, T.; Coman, I.L. Histopathological Lung cancer Detection using Enhanced Grasshopper Optimization Algorithm with Random Forest. *Int. J. Intell. Eng. Syst.* **2022**, *15*, 11–20.

68. Phankokkruad, M. Ensemble Transfer Learning for Lung Cancer Detection. In Proceedings of the 4th International Conferenceon Data Science and Information Technology, Shanghai, China, 23–25 July 2021; pp. 438–442.