Hybrid Deep Learning and Genetic Algorithm Approach for Detecting Keratoconus Using Corneal Tomography

Somayeh Ghasedi and Amr R. Abdel-Dayem*

School of Engineering and Computer Science, Laurentian University, Sudbury, Ontario, Canada
Email: sghasedi@laurentian.ca (S.G.); aabdeldayem@laurentian.ca(A.R.A.-D.)
*Corresponding author

Manuscript received July 31, 2024; revised August 7, 2024; accepted September 25, 2024; published January 21, 2025

***Abstract*—Nowadays, there are still significant challenges encountered in the accurate diagnosis of various eye diseases, such as Keratoconus (KCN) and cataracts. Early detection of Keratoconus is crucial in preventing its progression and ensuring the best treatment outcomes. Artificial Intelligence (AI) is being widely applied in ophthalmology through the training of deep learning networks for the early detection of eye diseases. This research presents a novel, integrated machine learning approach for diagnosing Keratoconus disease by combining feature extraction through Convolutional Neural Networks (CNN) with a Support Vector Machine (SVM) and Artificial Neural Network (ANN) for classification. Employing a multi-objective genetic algorithm, the method optimizes feature selection, aiming to minimize both diagnostic error and the number of features. The study utilizes a dataset of 5,152 ophthalmic images (1288 samples) categorized into Normal (476), Suspect (453), and Keratoconus (359) cases. Combining a Convolutional Neural Network (CNN) for feature selection with a Genetic Algorithm (GA) significantly improved diagnostic accuracy. Consequently, by focusing on the most relevant features of Keratoconus, the model achieved an impressive 98.63% accuracy for ANN classification with a genetic algorithm, and 98.13% for SVM classification with a genetic algorithm. The accuracy of the algorithm exceeded that of when SVM and ANN were used without the genetic algorithm, which were 97.53% and 96.9% respectively, underscoring the benefit of combining Artificial Neural Networks (ANNs) with Genetic Algorithms (GAs) in KC diagnosis. Implementing this model can assist physicians in more accurate Keratoconus detection, providing better predictions regarding patients' eye conditions, and offering timely treatment recommendations.**

***Keywords*—Keratoconus (KCN) eye disease, Convolutional Neural Networks (CNN), Support Vector Machine (SVM), Artificial Neural Network (ANN), Multi-objective Genetic Algorithm**

I. Introduction

In recent years, Machine Learning (ML) and Artificial Intelligence (AI) have significantly impacted healthcare, advancing the diagnosis and treatment of various diseases. AI algorithms have been successfully employed in the early detection of cancer, improving the accuracy of breast cancer screenings [1, 2], predicting cardiovascular conditions [3, 4], and assisting in diagnosing neurodegenerative diseases such as Alzheimer's [5–8]. These technologies enhance diagnostic precision, enabling clinicians to detect complex patterns in medical data that are often challenging for human interpretation. However, significant challenges remain in diagnosing various eye diseases, such as Keratoconus (KCN) and cataracts, with some patients unfortunately going blind due to delays in initiating their treatment. Accurate and early diagnosis of Keratoconus presents significant challenges, especially in detecting subclinical cases that show no visible signs or symptoms, making them difficult to distinguish from healthy eyes [9]. The diversity of patterns seen in topographic maps of Keratoconic eyes can be misinterpreted, necessitating the development of automatic quantitative image analysis and objective criteria for accurate identification [10]. Despite modern diagnostic tools, early stages of Keratoconus may still be misdiagnosed due to initial subtle symptoms, making accurate diagnosis time-consuming and demanding [11]. Timely intervention is crucial to mitigate the financial burden of treatment and prevent vision loss in severe cases [12]. Early detection can also avoid the need for complex interventions like corneal transplantation [13]. Improvements in Keratoconus detection techniques, particularly in early detection, can significantly enhance the management and prognosis of the condition [11]. The incorporation of Artificial Intelligence (AI), machine learning (ML), and Deep Learning (DL) technologies in ophthalmic settings shows promise for early detection and timely treatment of eye disorders [14]. However, challenges remain, including evaluating the reproducibility, accuracy, and reliability of these algorithms in clinical healthcare practice [14]. As AI, ML, and DL techniques advance, they may significantly contribute to the diagnostic and therapeutic progress in ophthalmology [14, 15]. Convolutional Neural Networks (CNNs) have emerged as effective tools for image recognition and classification, including the diagnosis of Keratoconus. Their direct feature extraction capabilities make them advantageous over other machine learning methods [15]. Machine learning algorithms, like CNNs, can be instrumental in identifying patterns and characteristics related to classification [16]. Despite recent advancements in AI, ML, and DL, challenges remain in diagnosing subclinical cases of Keratoconus, which often lack visible signs. This study aims to address these challenges by integrating machine learning and artificial intelligence to improve the early detection and treatment of Keratoconus. We propose a novel model that combines Convolutional Neural Networks (CNN) for feature extraction with Support Vector Machine (SVM) and Artificial Neural Network (ANN) for classification. Additionally, we focus on optimizing the feature extraction process using a Genetic Algorithm (GA) to balance the number of features and diagnostic accuracy.

This paper is organized as follows: Section II reviews methodologies for diagnosing Keratoconus and the role of AI and optimization algorithms in ophthalmology. Section III details the dataset and methodology of the proposed approach. Section IV presents the experimental results, evaluating the accuracy of feature selection and image classification methods. Finally, Section V discusses conclusions and

potential future research.

II. Literature Review

Keratoconus, marked by progressive corneal thinning and protrusion, presents a notable challenge for ophthalmologists, as early and accurate diagnosis is essential for initiating timely interventions and preventing vision loss. The increasing prevalence of Keratoconus highlights the urgent requirement for reliable diagnostic methods [9]. Technological advancements, particularly in machine learning and artificial intelligence (AI), show promise in aiding ophthalmologists in early and reliable disease diagnosis. Recognizing the importance of diagnosing Keratoconus in its early stages, efforts have been made to develop automated diagnostic systems using AI, machine learning, and deep learning technologies. These technologies demonstrate their capability in diagnosing Keratoconus [12, 17–19]. On the other hand, classification algorithms play a crucial role in medical diagnosis by offering precision, efficiency, and consistency in detecting eye disorders. They enable early detection, leading to timely intervention and improved patient outcomes. By analyzing extensive medical data and images, classification algorithms provide accurate and consistent diagnoses while automating image analysis, thus saving time for both eye specialists and patients. Moreover, they support informed decision-making for treatment planning and can be combined with other machine learning methods to enhance diagnostic accuracy [11, 20].

A. The Use of SVM Classification Algorithms in Diagnosis of KCN Eye Diseases

Support Vector Machine (SVM) stands out as a highly effective tool in diagnosing Keratoconus (KCN) and related eye diseases. The robustness of SVM in handling outliers and high-dimensional data, along with its flexibility in adapting to various data distributions through different kernel functions, underscores its significance in medical imaging [21, 22]. SVM has found extensive application in analyzing corneal images and parameters, facilitating the creation of automated diagnostic models. Notably, SVM excels in feature selection, enhancing diagnostic accuracy while streamlining model complexity [21]. Several studies underscore the effectiveness of Support Vector Machines (SVM) in diagnosing Keratoconus.

Al-Timemy *et al.* [23] employed a hybrid deep learning and SVM model, achieving 97.7% accuracy for a two-class problem and 84.4% for a three-class problem, revealing the model's limitations in more complex classifications. Arbelaez *et al.* [21] demonstrated SVM's exceptional accuracy (>95%) in distinguishing between normal, Keratoconus, and subclinical Keratoconus eyes using Scheimpflug camera and Placido corneal topography data. Similarly, Mosa *et al.* [24] and Gao *et al.* [22] utilized SVM for high accuracy and specificity in Keratoconus detection, highlighting the model's robustness in handling diverse datasets. Toutounchian *et al.* [25], and Souza *et al.* [26] further confirmed SVM's high diagnostic accuracy using topographical maps, while Del Río *et al.* [27] compared SVM with other classifiers, demonstrating its superior performance. These studies collectively highlight SVM's pivotal role in enhancing the precision of Keratoconus diagnosis through advanced feature extraction and classification techniques.

B. The Use of ANN Classification Algorithms in Diagnosis of KCN Eye Diseases

In diagnosing Keratoconus, ANNs are essential for analyzing corneal images and detecting subtle patterns that may elude human observers [12]. Moreover, ANNs excel in automatically extracting pertinent features from medical images, thereby enhancing classification accuracy. However, their efficacy hinges on access to large and diverse datasets for robust training and validation, facilitating model generalization. Metrics such as accuracy, sensitivity, specificity, and AUC serve as benchmarks to evaluate the effectiveness of ANN models in diagnosing Keratoconus [16, 23]. Despite their potential, the reliance of ANNs on extensive and high-quality datasets for training poses a challenge, as such datasets may not always be readily available.

Mehdizadeh Dastjerdi *et al.* [17] highlighted the practical implications of ANNs in clinical settings, aiding ophthalmologists in selecting suitable surgery candidates. Kovacs *et al.* [28] demonstrated ANNs' efficacy in early Keratoconus detection, achieving higher precision than traditional methods. Elsawy *et al.* [18], developed Multidisease Deep Learning Neural Networks (MDDN) that accurately diagnose corneal diseases such as Keratoconus using AS-OCT images, achieving high accuracy (AUROCs >0.99, AUPRCs >0.96, F1 scores >0.90). Additionally, innovative approaches utilizing numerical computing techniques integrating feed-forward ANN and optimization technique, as proposed by Umar *et al.* [29] and combining Gudermannian Neural Network (GNN) with a hybrid optimization approach, as demonstrated by Sabir *et al.* [30], underscore the versatility and effectiveness of ANN-based methodologies in solving complex corneal shape models in eye surgery.

C. The Use of CNN Architecture Algorithms in Diagnosis of eye Diseases

CNN architecture algorithms play a pivotal role in diagnosing eye diseases by automatically learning and extracting features from medical images such as corneal and retinal scans. Popular CNN architectures like VGG16, InceptionV3, and ResNet152 have demonstrated high accuracy, sensitivity, and specificity in identifying conditions like Keratoconus [23, 31, 32]. Techniques like transfer learning allow pre-trained CNN models to be fine-tuned for specific tasks, enabling early detection and timely intervention [18, 23]. Combining CNNs with other machine learning methods like Support Vector Machines (SVMs) and Artificial Neural Networks (ANN) forms hybrid approaches that leverage the strengths of different algorithms, thereby improving diagnostic accuracy [19]. Numerous studies have demonstrated the efficacy of CNN-based models in diagnosing Keratoconus. Kou *et al*. [33] found that CNN models like VGG16, InceptionV3, and ResNet152 achieved high sensitivity and specificity. Al-Timemy *et al.* [13] used an ensemble of pre-trained CNN networks, including SqueezeNet, AlexNet, ShuffleNet, MobileNet-v2, achieving 98.3% accuracy, though their approach lacked optimization techniques. In another study, Al-Timemy *et al.* [19] used Xception and InceptionResNetV2 to extract features from topography maps, while omitting Elevation Back and

Pachymetry images in the three-class problem, which makes the final model not fully reliable. Kamiya *et al.* [15] showed VGG-16's effectiveness in distinguishing Keratoconus from normal eyes with high accuracy. Lavric *et al.* [16] introduced KeratoDetect, a CNN-based algorithm with 99.33% accuracy. Feng *et al.* [12] developed KerNet, using CNN-based cascaded residual blocks, achieving superior diagnostic accuracy, highlighting CNN's pivotal role.

D. The Use of Optimization Techniques for Feature Selection

The integration of optimization techniques such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) with classification methods plays a crucial role in diagnosing eye diseases from images [34]. These algorithms automate feature selection, reducing data dimensionality, and enhancing the speed and accuracy of classification tasks. For instance, Feng *et al.* [12] introduced Cartesian Genetic Programming (CGP) to automatically design competitive Convolutional Neural Network (CNN) architectures, minimizing the need for expert knowledge and extensive trial and error. This innovative approach optimizes CNN structure and connectivity using CGP encoding, maximizing validation accuracy. Similarly, Subramanian and Ramesh [20] emphasized the significance of Particle Swarm Optimization (PSO) in selecting relevant indices for Keratoconus diagnosis, aiming to reduce computing time and errors in computerized diagnosis. By focusing on segmenting topography images, PSO and its modifications improve performance metrics such as accuracy, sensitivity, and specificity, thereby advancing Keratoconus diagnosis and classification. Other research studies have applied the integration of optimization techniques with classification algorithms to various medical applications. For example, Li *et al.* [35] utilized a genetic Algorithm (GA) to refine the initial weights of a Convolutional Neural Network (CNN) for liver CT tumor classification. By leveraging GA's global optimization capabilities, the method generated an optimal set of initial weights, resulting in improved learning performance compared to traditional CNNs. Similarly, Llorella *et al.* [36] and Kabir Anaraki *et al.* [37] employed GA to optimize CNN network parameters, enhancing image classification quality in Brain-Computer Interface (BCI) systems and noninvasively classifying different grades of glioma brain tumors using magnetic resonance imaging (MRI), respectively. In both studies, GA significantly reduced computation time for architecture selection while improving classification accuracy. Additionally, Davoudi and Thulasiraman [38], and Rodrigues *et al.* [39] used GA to optimize CNN architectures for various tasks, achieving superior performance and reduced computation time.

Building on the review of past studies, this research aims to develop a new model that integrates a Convolutional Neural Network (CNN) for feature extraction with Support Vector Machine (SVM) and Artificial Neural Network (ANN) models for detecting Keratoconus (KCN). The feature extraction process will focus on two main objectives: optimizing the number of features and increasing accuracy using a Genetic Algorithm (GA). This integrated approach aims to enable early detection and intervention, which are crucial for effective treatment and improved patient outcomes.

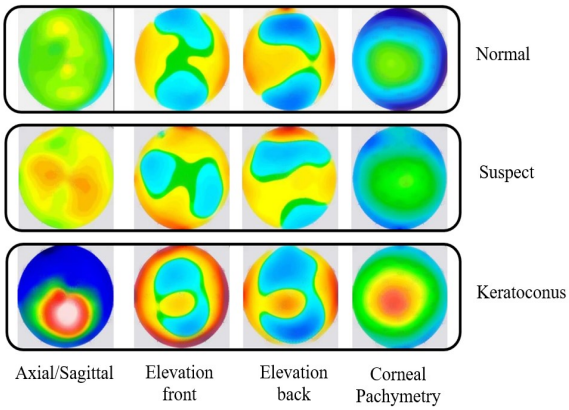

Fig. 1. Samples of topography images used in this study.

III. Materials and Methods

A. Dataset Organization and Feature Extraction

This study utilizes a dataset consisting of ophthalmic images sourced from an eye clinic in Egypt, previously employed in a study conducted by Al-Timemy *et al.* [19]. The dataset used in this study comprises 5152 images from 1288 eyes (samples), including 476 Normal, 453 Suspect, and 359 Keratoconus cases. Each sample comprises four images, including Axial/Sagittal curvature, Pachymetry, Elevation front, and Elevation back, containing specific eye-related data, and the dimensions of the images are 256×256 pixels. Fig. 1 shows samples of the 4 corneal maps for the 3 classes investigated in this study.

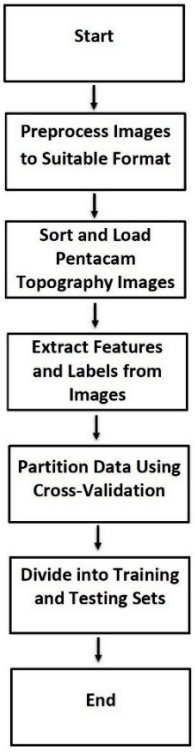

Fig. 2. Data processing workflow for Pentacam topography images.

In corneal topography analysis, color variations provide crucial insights into corneal health. Blue areas in Pentacam topography maps indicate corneal thickness variations, with minimal blue in healthy corneas and extensive blue in advanced Keratoconus (KCN), signifying severe thinning and steepening. Purple regions suggest localized thinning,

potentially indicating conditions like pellucid marginal degeneration. Orange coloration, minimal in healthy corneas, suggests irregularities in suspect and KCN cases, reflecting significant steepening and thinning. Predominant green signifies uniform thickness and regular curvature, while yellow indicates potential early irregularities. Red areas highlight minor irregularities in healthy corneas and severe thinning in KCN. White regions indicate severe KCN or other conditions like scarring or post-surgical changes, necessitating comprehensive evaluation. Symmetrical color distributions denote normal corneal shape, while asymmetrical patterns in suspect and KCN cases indicate underlying irregularities requiring further diagnostic testing [40–42].

Fig. 2 shows the block diagram of the data processing workflow for Pentacam topography images. The subsequent sections will present the processing stages in more detail.

1) Preprocessing techniques for Pentacam topography images

The dataset for this study comprises high-resolution Pentacam topography images, capturing the eye's anterior segment, particularly the cornea, with great precision. Rigorous pre-processing steps were undertaken to ensure the dataset's integrity and reliability. During the initial examination, some images were found with incomplete topographical depictions due to potential imaging limitations. These deficient images were systematically excluded to maintain dataset accuracy and prevent biases. For feature scaling and transformation, since the AlexNet architecture is used in this study for feature extraction from images, and this architecture requires input images to be 227×227 pixels to effectively perform convolution and pooling operations necessary for feature extraction, the topography images from Pentacam devices were resized. This preprocessing step ensures that the images are scaled to be compatible with AlexNet, enabling consistent and efficient processing across all inputs, which is critical for the success of tasks like classification or anomaly detection within the network. The original images, sized at 256×256 pixels, were resized to 227×227 pixels—a reduction of less than ten percent per dimension. This small resizing preserved the number of data layers and did not compromise the detailed corneal measurements, ensuring the images remained accurate and reliable for analysis. Regarding data cleaning procedures, the advanced Pentacam imaging technology produces high-quality, reliable data, negating the need for additional processing like filtering or noise removal. The technology ensures the data are ready for immediate analysis, maintaining the highest quality. Overall, these meticulous pre-processing steps maintain the dataset's integrity, supporting accurate and reliable analysis. This rigorous approach enhances the robustness and validity of the research findings, contributing to the practical effectiveness of the SVM and ANN models trained with the genetic algorithm.

2) Image dataset organization and loading procedure

The imageDatastore function in MATLAB efficiently handles and preprocesses image datasets, creating a labeled datastore. It reads images from specified folders ('Keratoconus,' 'Normal,' 'Suspicious') in a consistent order, facilitating organized management and processing. In this study, four types of topography images (Axial, Elevation Front, Elevation Back, and Pachymetry) are categorized into three groups (Keratoconus, Normal, Suspicious) using MATLAB's imageDatastore function. For each topography image, such as Axial, each subfolder contains images corresponding to its label, which are utilized for training the neural network. The labels for the images are automatically assigned based on the folder names (Keratoconus, Normal, Suspicious). This process is also carried out for the remaining topography images (Elevation Front, Elevation Back, and Pachymetry) so that after saving the images and labels, the features related to the images can be extracted in the next step.

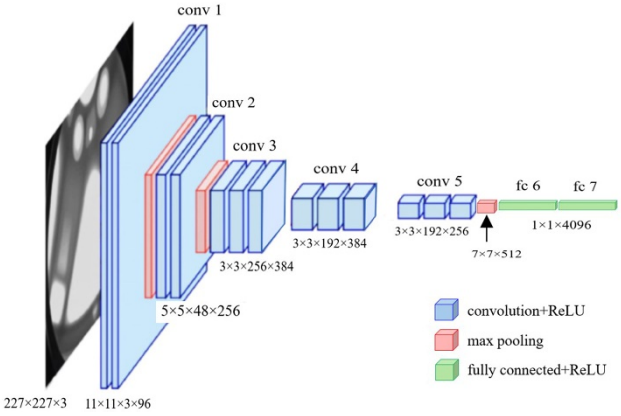

Fig. 3. Training images process using CNN (AlexNet Architecture).

3) Feature extraction using AlexNet Architecture

The architecture utilized in this study is the AlexNet Architecture, one of the Convolutional Neural Network (CNN) models designed specifically for image processing [43]. The structure of the CNN consists of two main parts: Feature Extraction and Classification. The CNN network receives Pentacam topography images as input, in the form of an array of pixels, and then CNN performs the necessary preprocessing on images using its architecture. As an illustration, the model receives 2D topography scans as an input and then AlexNet Architecture is used as an encoder to extract the main features which includes multiple convolutional layers, max-pooling layers followed by fully connected layers which is a vector with a size of 4096. Fig.3 provides the overview of the proposed CNN architecture utilized in this study.

4) Cross validation

In this study, cross-validation technique is employed to mitigate overfitting by dividing the dataset into training and testing sets. The model is trained on one subset and tested on another, with this process repeated multiple times to ensure robust performance evaluation. Specifically, a 3-fold cross-validation is implemented, dividing the dataset into three equal sets, each containing an even distribution of Normal, Suspect, and Keratoconus (KCN) cases. Each set includes data from four types of topography images (Axial, Elevation Front, Elevation Back, and Pachymetry), ensuring comprehensive training and evaluation. The final setup includes 24 variables representing features and labels across all sets and image types. To form the first cross-validation set, the labels for the first validation set, which consists of 429 rows across all groups (Axial, Elevation Front, Elevation

Back, and Pachymetry), must be checked. In these 429 rows, for each topography image, if the image in the first set of cross-validation maintains the same label (Keratoconus, Normal, or Suspect) across all groups (Axial, Elevation Front, Elevation Back, and Pachymetry), that case is classified accordingly. The features from the four topography images (Axial, Elevation Front, Elevation Back, and Pachymetry) are then combined, creating the features and labels for the first cross-validation set, which constitutes one-third of the data. Consistent labeling across all topography types is ensured for each selected case. Thus, the feature vector changes from 4096 for each image to 16384 for four images with an accompanying label. The same process is carried out for the other two cross-validation sets. Finally, the three main sets are saved in a separate file to be loaded in the next step for performing the testing and training operations

B. Data Classification and Feature Optimization

This step involves data classification using Support Vector Machine (SVM) and Artificial Neural Network (ANN) based on features extracted from AlexNet. Initially, SVM and ANN classify the data using these extracted features. Then, a genetic algorithm optimizes the features, and the data is reclassified using SVM and ANN with the optimized features. The process begins by using SVM to classify topographic images from the available datasets. The features of each topographic image are input into the SVM, which outputs classifications and groups samples into three categories: Normal, Suspect, and Keratoconus. Notably, each sample is characterized by 16,834 extracted features, derived from 4 topographic images per case. Next, classification using an ANN with 6 hidden neurons is performed to group the samples, utilizing all available features. Finally, a genetic algorithm optimizes all features, and the optimized features are employed for classification, first using SVM and then refined by ANN techniques. The following subsections will detail the implementation of the genetic algorithm for feature optimization and highlight the performance evaluation method.

1) Feature optimization via Genetic Algorithm

A Genetic Algorithm is employed to randomly create an initial population comprising 100 generations. This is represented by a matrix with 100 rows and 16,384 columns, where each row (chromosome) contains binary values indicating the presence (1) or absence (0) of specific features extracted from the images using the AlexNet architecture. The randomness introduced by this process allows the GA to explore diverse combinations of features, which is essential for optimizing the feature selection process. In the feature subset selection phase, the final layer of AlexNet, the fc7 layer, extracts a 4096-dimensional feature vector for each image. Four types of Pentacam images (Axial, Elevation Front, Elevation Back, and Pachymetry) are processed, resulting in a combined feature vector of 16,384 elements per sample. The initial population's chromosomes guide which features are selected, using a binary vector ('dvar') to indicate feature inclusion or exclusion. This vector enables the GA to experiment with various feature combinations, aiming to optimize classification accuracy for KCN, Normal, and Suspect cases. The classification process employs both SVM and ANN models, trained on features selected by the GA. The dataset is divided into three subsets for cross-validation. Each subset rotates as the test set while the other two subsets are used for training. This process ensures each subset is tested exactly once, facilitating robust model evaluation.

The MATLAB function 'fitcecoc' configures and trains the SVM multiclass classification model using a One-vs-All strategy, breaking the multi-class problem into binary classification tasks. Each classifier distinguishes one class from the others, and the class with the highest confidence score is chosen as the final prediction. For ANN training, the MATLAB function 'patternnet' initializes the network, designed with an input layer matching the GA-selected features, six hidden neurons, and an output layer with three neurons for each class. The ANN training process divides the data into training, validation, and test sets (70%, 15%, and 15%, respectively). The neural network learns by adjusting its settings to minimize errors, enhancing its ability to classify new data correctly. The ANN pairs input features with target labels to learn associations, crucial for accurate classification. During the ANN evaluation, the transposed test features are combined with the training features into a single matrix, allowing the neural network to predict on both datasets simultaneously. The Softmax activation function in the output layer generates a probability distribution across classes, selecting the class with the highest probability as the final classification. Each image is assigned a label based on these probabilities, ensuring precise categorization into Normal, Suspect, or KCN.

2) Performance evaluation of SVM and ANN Classification models optimized with Genetic Algorithm

In this stage of the study, the performance of the Support Vector Machine (SVM) and Artificial Neural Network (ANN) classifiers is evaluated by calculating metrics such as training error and the number of selected features. The error calculation is performed by comparing the true labels from both the training and test sets with the predicted labels generated by the classifiers. For the SVM, the classifier predicts class labels for each image, assigning a vector indicating membership in one of the classes: Normal, Suspect, or Keratoconus. The classification error is computed by counting the mismatches between the true and predicted labels. Each mismatch is marked as an error, and the total number of mismatches provides a measure of the classifier's accuracy. The same process is applied for ANN, with a slight modification due to its nature of counting each mismatch twice, necessitating division by two to correct for the dual comparisons per data point. The ANN also predicts labels in a one-hot encoded format, comparing each predicted element with the actual label to identify mismatches, which are summed to calculate the prediction error. The number of selected features is determined by a binary vector, dvar, where each element indicates whether a feature is selected. The sum of these values gives the total number of features used for training the classifier. This process is essential for understanding feature selection's impact on model performance.

Both SVM and ANN models undergo three rounds of evaluation using cross-validation sets. A confusion matrix is generated to compare actual and predicted labels, providing a comprehensive analysis of the classifier's performance. The average of the three error rates is reported as the final error, and the number of features selected by the genetic algorithm is noted. The genetic algorithm optimizes feature selection by

minimizing the classification error and the number of features. The fitness function evaluates each model, balancing these two objectives. A Pareto diagram is used to visualize this balance, with the X-axis representing the error rate and the Y-axis showing the number of features. Points on the Pareto front highlight the best trade-offs with low error and fewer features. The genetic algorithm evaluates each chromosome based on these metrics, retaining those close to the Pareto front for further generations while discarding suboptimal ones. Some suboptimal chromosomes are maintained to preserve genetic diversity. Offspring are generated through crossover and mutation to explore new feature combinations. Over successive generations, the population evolves toward more optimal solutions, moving the Pareto front closer to the origin, indicating improved performance. In conclusion, this iterative evaluation and optimization process ensures continuous improvement in the model's accuracy and feature efficiency. The final model, optimized by the genetic algorithm, achieves a specific level of average error and minimal features, demonstrating the effectiveness of combining SVM and ANN classifiers with genetic algorithms for early detection of Keratoconus.

IV. Experimental Results

The evaluation of the proposed methods aims to find an effective subset of features for SVM and ANN classification for Keratoconus patients. In other words, through the evaluation of the proposed methods, it can be inferred that the highest accuracy in the classification and prediction of Keratoconus patients is achieved by these methods that select optimal features using the genetic algorithm for classification. Therefore, for the evaluation of the proposed methods, a comparison is made between the predicted class labels for test samples in the Keratoconus patient dataset and the actual class labels declared for these samples in the original dataset. However, it is evident that this classification problem is not limited to only two classes, and the implementation of the Confusion Matrix for Multiclass Classification is required. To generate the Confusion Matrix, the data have been entered as shown in Table 1.

Table 1. Model input data

	KCN (real)	NORMAL (real)	SUSPECT (real)
KCN (prediction)	Patients with Keratoconus correctly predicted as Keratoconus	Normal patients mistakenly predicted as Keratoconus	Suspect patients mistakenly predicted as Keratoconus
NORMAL (prediction)	Patients with Keratoconus mistakenly predicted as Normal	Normal patients correctly predicted as Normal	Suspect patients mistakenly predicted as Normal
SUSPECT (prediction)	Patients with Keratoconus mistakenly predicted as Suspect	Normal patients mistakenly predicted as Suspect	Suspect patients correctly predicted as Suspect

In each stage, the target case is considered as the positive class, and other cases are considered as negative classes. To investigate and evaluate the proposed method and provide a comparison, a confusion matrix is used, and the summary of the results of this comparison is categorized into four categories, including TRUE POSITIVE, TRUE NEGATIVE, FALSE POSITIVE, and FALSE NEGATIVE. Table 2 describes the details of these indexes for each type of image class.

The evaluation process utilizes the confusion matrix for each class. Averaging the precision of these metrics across all classes provides a general idea of the model's performance. Precision, specifically, is calculated as the number of True Positives divided by the sum of True Positives and False Positives for a particular class. These evaluation metrics, derived from the confusion matrix parameters, serve as a tool for assessing the quality of the proposed method and comparing it with existing methods. Therefore, in the following, the confusion matrices and sensitivity analysis of the proposed feature selection method based on the genetic algorithm for Support Vector Machine (SVM) and Artificial Neural Network (ANN) classification with the standard SVM and ANN classification approaches have been investigated separately.

Table 2. Main indexes of confusion matrix

	KCN	NORMAL	SUSPECT
TURE POSITIVE (TP)	The number of KCN patients correctly considered as KCN	The number of Normal patients correctly considered as Normal	The number of Suspect patients correctly considered as Suspect
FALSE POSITIVE (FP)	The number of non-KCN patients mistakenly considered as KCN	The number of non-Normal patients mistakenly considered as Normal	The number of non-Suspect patients mistakenly considered as Suspect
FALSE NEGATIVE (FN)	The number of KCN patients mistakenly considered in other groups	The number of Normal patients mistakenly considered in other groups	The number of Suspect patients mistakenly considered in other groups
TRUE NEGATIVE (TN)	The number of non-KCN patients correctly considered as non-KCN	The number of non-Normal patients correctly considered as non-Normal	The number of non-Suspect patients correctly considered as non-Suspect

A. Confusion Matrix for the Accuracy of the Algorithm Using ANN

Fig. 4 shows the confusion matrix of the created neural network, representing the accuracy of the network in the prediction of KCN, Suspect and Normal cases. The neural network used in this study utilizes the cross-validation technique, which includes training and testing data divided into three subsets. In the first step, the first subset of data is used as testing data and the remaining data is used for training.

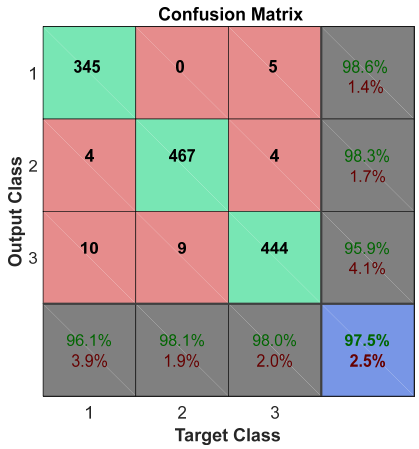

Fig. 4. Confusion Matrix for the first subset of cross validation (ANN).

As shown in Figs. 4 to 6, the squares marked in green —

(1,1), (2,2), and (3,3) — represent samples correctly predicted by the network. Conversely, the squares highlighted in red — (1,2), (1,3), (2,1), (2,3), (3,1), and (3,2) — indicate cases where the network made incorrect predictions. The percentages in the gray squares at the bottom of the confusion matrix represent recall metric.

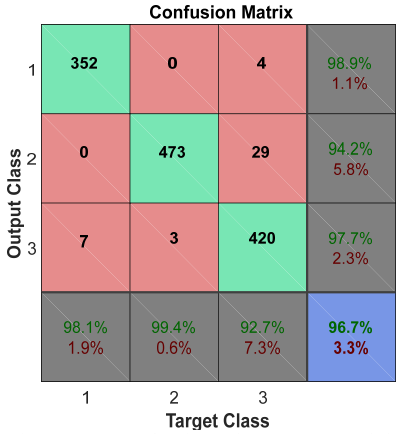

Fig. 5. Confusion Matrix for the first subset of cross validation (ANN).

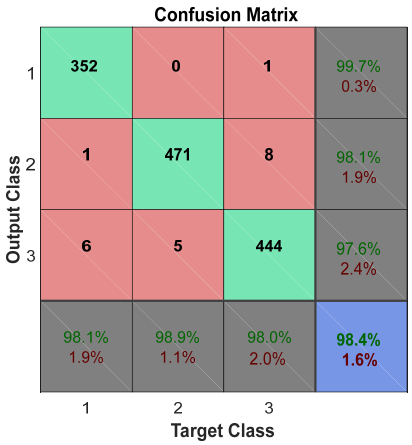

Fig. 6. Confusion Matrix for the third subset of cross validation (ANN).

Eq. (1) shows that recall is calculated by dividing the number of true positive predictions by the total number of actual positives, which includes both true positives and false negatives. The percentages in the gray squares to the right of the confusion matrix represent precision. Eq. (2) shows that precision is calculated by dividing the number of true positive predictions by the total number of predicted positives, which includes both true positives and false positives. Eq. (3) represents the F1 score, which is the harmonic mean of precision and recall. It accounts for both false positives and false negatives, making it particularly effective for handling imbalanced datasets. Finally, the percentage displayed in the blue square at the bottom right corner of the confusion matrix represents the overall accuracy of the model. Eq. (4) shows that accuracy is calculated by dividing the total number of correct predictions (the sum of the diagonal elements, which correspond to the true positives for each class) by the total number of predictions made (the sum of all elements in the matrix).

$$\text{Recall} = TP / (TP + FN) \quad (1)$$

$$\text{Precision} = TP / (TP + FP) \quad (2)$$

$$\text{F1 Score} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (3)$$

$$\text{Accuracy} = \sum \text{Diagonal Elements} / \sum \text{All Elements} \quad (4)$$

The results obtained from running the SVM and ANN algorithms without using a genetic algorithm will yield a point with a specific number of errors and features, which is not optimal. To calculate the algorithm's accuracy, the average confusion matrix across a three-fold cross-validation will be computed.

To be more specific, as shown in Fig. 4, in the first step, 345 samples (1380 images) out of 359 samples (1436 images) related to KCN patients were correctly predicted. However, 4 KCN cases (16 images) were mistakenly diagnosed as Normal, and 10 KCN cases (40 images) were mistakenly diagnosed as Suspicious. Similarly, for Normal cases, 467 samples (1868 images) out of 476 cases (1904 images) were correctly predicted. No Normal case was mistakenly diagnosed as KCN, but 9 Normal cases (36 images) were mistakenly diagnosed as Suspicious. Lastly, 444 samples (1776 images) out of 453 samples (1812 images) corresponding to Suspicious patients were correctly predicted, while 5 Suspicious cases (20 images) were mistakenly diagnosed as KCN, and 4 Suspicious cases (16 images) were mistakenly considered as Normal.

On the other hand, considering the percentages in the gray squares at the far right of the plot, 98.6% precision for Class 1 (KCN) indicates that out of all instances the model predicted to be KCN, 98.6% were actually KCN. Similarly, 98.3% precision for Class 2 (Normal) means that out of all the instances the model predicted to be Normal, 98.3% were Normal. Furthermore, the 95.9% precision for Class 3 (Suspicious) suggests that out of all instances predicted as Suspicious by the model, 95.9% truly were Suspicious. Overall, the prediction accuracy of this step turned out to be 97.5%. In the second and third steps, the same process is followed: first, the second subset of data is used as testing data while the remaining data is used for training, and then the third subset of data is used as testing data while the remaining data is used for training. The confusion matrices for the second and third subset of data are presented in Fig. 5 and Fig. 6, which show 96.7% and 98.4% prediction accuracy. The observed fluctuations in accuracy result from variations in the test and training datasets. Instead of considering a singular accuracy value, the mean accuracy across all values is calculated, resulting in a mean accuracy of 97.53%.

Additionally, the final error of the ANN algorithm after three-fold cross-validation is calculated using the formula below, yielding an error of 32 with 16384 features.

$$\frac{(1288 \times 3) - (444 + 471 + 352 + 420 + 473 + 352 + 345 + 467 + 444)}{3} = 32$$

B. Confusion Matrix for the Accuracy of the Algorithm Using SVM Classification

As shown in Fig. 7, in the first subset, 352 samples (1408 images) out of 359 samples (1436 images) related to KCN patients are correctly predicted. One KCN case (4 images) is mistakenly diagnosed as a Normal case, and 6 KCN cases (24 images) are mistakenly diagnosed as suspicious. Similarly, for Normal cases, 471 (1884 images) out of 476 cases (1904 images) are correctly predicted. No Normal cases have mistakenly been diagnosed as KCN but 5 Normal cases (20 images) are mistakenly diagnosed as suspicious. Finally, 441 samples (1764 images) out of 453 samples (1812 images) corresponding to suspicious patients are correctly predicted,

while 4 suspicious cases (16 images) are mistakenly diagnosed as KCN, and 8 suspicious cases (32 images) are mistakenly considered as Normal. On the other hand, considering the percentages in the gray squares at the far right of the plot, 98.9% precision for Class 1 (KCN) indicates that out of all instances the model predicted to be KCN, 98.9% were actually KCN. Similarly, 98.1% precision for Class 2 (Normal) means that out of all the instances the model predicted to be Normal, 98.1% were Normal. Furthermore, the 97.6% precision for Class 3 (Suspicious) suggests that out of all instances predicted as Suspicious by the model, 97.6% truly were Suspicious. Overall, the prediction accuracy of this step turned out to be 98.1%. The confusion matrices for the second and third subset of data are presented in Fig. 8 and Fig. 9, which show 97.6% and 95% prediction accuracy, resulting in a mean accuracy of 96.90% across three matrices.

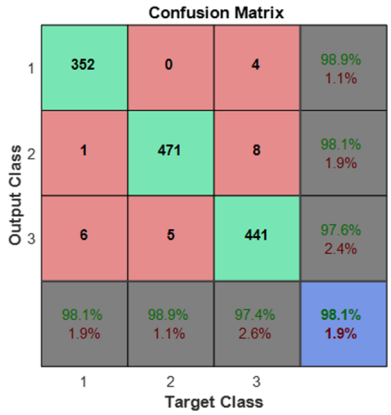

Fig. 7. Confusion Matrix for the first subset of cross validation (SVM)

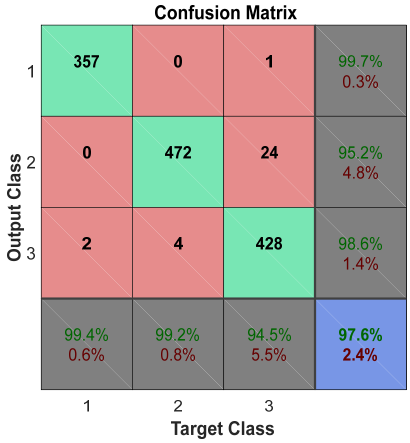

Fig. 8. Confusion Matrix for the second subset of cross validation (SVM).

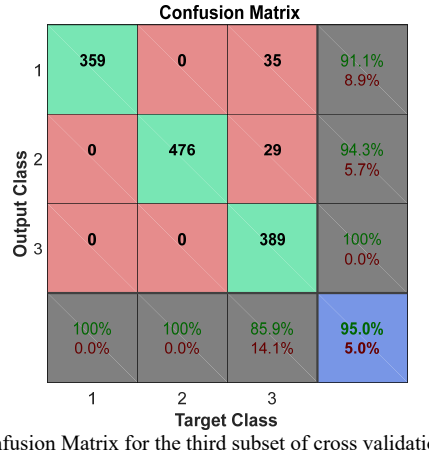

Fig. 9. Confusion Matrix for the third subset of cross validation (SVM).

The final error of the SVM algorithm after three-fold cross-validation is calculated using the formula below, yielding an error of 39 with 16384 features.

$$\frac{(1288 \times 3) - (359 + 389 + 476 + 428 + 472 + 357 + 352 + 471 + 441)}{3} = 39.66$$

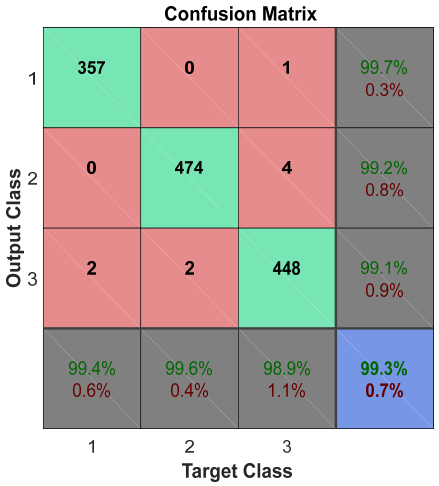

Fig. 10. Confusion Matrix for the first subset of cross validation- (ANN and GA).

C. Confusion Matrix of Algorithm Accuracy Using ANN Classification and GA for Feature Selection

Reducing feature size from 16384 to 7103 by utilizing a genetic algorithm for feature selection lowers classifier complexity, cuts computational load and training time, and reduces overfitting, improving ANN efficiency. As shown in Fig. 10, in the first subset, 357 samples (1428 images) out of 359 samples (1436 images) related to KCN patients have been correctly predicted. On the other hand, no KCN cases have mistakenly been diagnosed as Normal, and 2 KCN cases (8 images) are mistakenly diagnosed as suspicious. Similarly, for Normal cases, 474 (1896 images) out of 476 cases (1904 images) of Normal patients are correctly predicted. Additionally, no Normal cases have mistakenly been diagnosed as KCN cases, and 2 Normal cases (8 images) are mistakenly diagnosed as suspicious. Finally, 448 samples (1792 images) out of 453 samples (1812 images) corresponding to suspicious patients are correctly predicted, while one suspicious case (4 images) is mistakenly considered as KCN, and 4 suspicious cases (16 images) are mistakenly considered as Normal. On the other hand, considering the percentages in the gray squares at the far right of the plot, 99.7% precision for Class 1 (KCN) indicates that out of all instances the model predicted to be KCN, 99.7% were actually KCN. Similarly, 99.2% precision for Class 2 (Normal) means that out of all the instances the model predicted to be Normal, 99.2 % were Normal. Furthermore, the 99.1% precision for Class 3 (Suspicious) suggests that out of all instances predicted as Suspicious by the model, 99.1% truly were Suspicious. Overall, the prediction accuracy of this step turned out to be 99.3%. The confusion matrices for the second and third subset of data are presented in Fig 11 and 12, which show 97.7% and 98.9% prediction accuracy, resulting in a mean accuracy of 98.63% across three matrices. Additionally, the optimum point of the algorithm with a minimum error of 17 and 6994 features is reported.

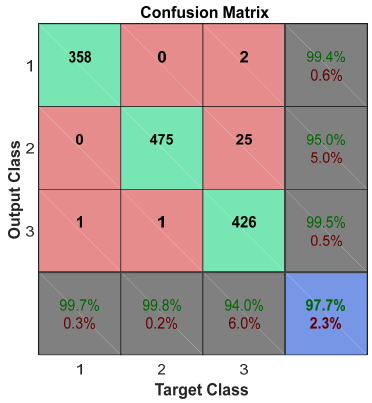

Fig. 11. Confusion Matrix for the second subset of cross validation- (ANN and GA).

The reported results indicate that the algorithm's accuracy is consistent across all three subsets (99.3%, 97.7%, and 98.9%). This suggests that the neural network operates uniformly and reasonably in this scenario. The results show that using a genetic algorithm for feature selection helps eliminate features that create noise or mislead the ANN. By reducing the number of features, the accuracy of the algorithm increases. Consequently, our network becomes more organized, and with less disturbance, classification accuracy increases. However, using the genetic optimization algorithm in this method essentially eliminates features that complicate the ANN network or create disruptions in the response, leading to incorrect predictions or responses. Therefore, feature selection with Genetic algorithm helps to achieve better results with ANN.

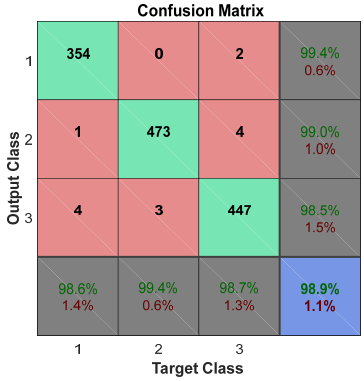

Fig. 12. Confusion Matrix for the third subset of cross validation- (ANN and GA).

D. Confusion matrix of algorithm accuracy using SVM classification and GA for feature selection

Reducing feature size from 16384 to 6999 by utilizing a genetic algorithm for feature selection lowers classifier complexity, cuts computational load and training time, and reduces overfitting, improving SVM efficiency. As can be seen in Fig. 13, 352 samples (1408 images) out of 359 samples (1436 images) related to KCN patients have been correctly predicted. On the other hand, 1 KCN case (4 images) is mistakenly diagnosed as Normal, and 6 KCN cases (24 images) are mistakenly diagnosed as suspicious. Similarly, for Normal cases, 471 (1884 images) out of 476 cases (1904 images) of Normal patients are correctly predicted. Additionally, no Normal cases have mistakenly been diagnosed as KCN cases, and 5 Normal cases (20 images) are mistakenly diagnosed as suspicious. Finally, 445 samples (1780 images) out of 453 samples (1812 images) corresponding to suspicious patients are correctly predicted, while 2 cases of suspicious cases (8 images) have mistakenly been diagnosed as KCN cases, and 6 suspicious cases (24 images) are mistakenly considered as Normal. On the other hand, considering the percentages in the gray squares at the far right of the plot, 99.4% precision for Class 1 (KCN) indicates that out of all instances the model predicted to be KCN, 99.4% were actually KCN. Similarly, 98.5% precision for Class 2 (Normal) means that out of all the instances the model predicted to be Normal, 98.5% were Normal. Furthermore, the 97.6% precision for Class 3 (Suspicious) suggests that out of all instances predicted as Suspicious by the model, 97.6% truly were Suspicious. Overall, the prediction accuracy of this step turned out to be 98.40%. The confusion matrices for the second and third subset of data are presented in Fig. 14 and Fig. 15, which show 97.8% and 98.2% prediction accuracy, resulting in a mean accuracy of 98.13% across three matrices. Additionally, the optimum point of the algorithm with a minimum error of 17 and 6994 features is reported.

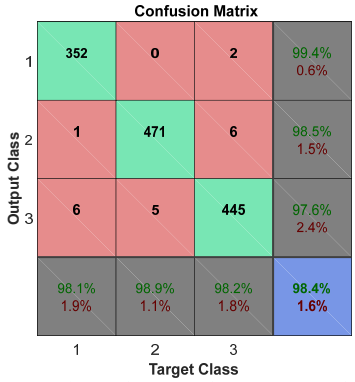

Fig. 13. Confusion Matrix for the first subset of cross validation (SVM and GA).

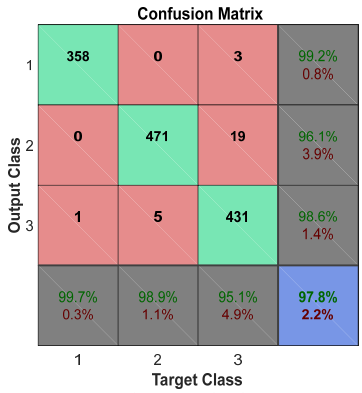

Fig. 14. Confusion Matrix for the second subset of cross validation (SVM and GA).

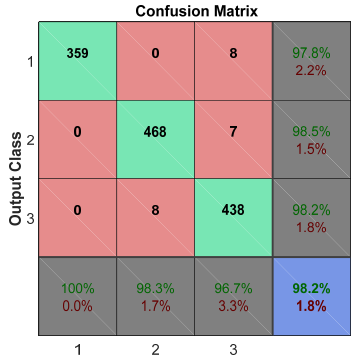

Fig. 15. Confusion Matrix for the third subset of cross validation (SVM and GA).

The reported results indicate that the algorithm's accuracy is consistent across all three subsets (98.4%, 97.8%, and 98.2%). This suggests that the SVM operates uniformly and reasonably in this scenario, similar to ANN.

The results indicate that while feature selection with the genetic algorithm improves the accuracy of both SVM and ANN classifications, ANN combined with the genetic algorithm achieves higher accuracy than SVM with the genetic algorithm.

E. Determination of optimal point selection for SVM and ANN classifications with Genetic Algorithm

The result of the combination of SVM and ANN with the optimization process is visualized using a Pareto diagram (Fig. 16 and Fig. 17), which helps to understand the trade-offs between classification error and the number of features selected by each solution. For each optimal point identified in the Pareto diagrams and their related tables (Table 3 and Table 4), the corresponding optimal feature vector is reported. This allows for analysis of which features are effective in detecting Pentacam topography images at each specific optimal point, and conversely, which features are ineffective. To select the best optimum point, among these points, the one with the least Euclidean distance from the origin (0,0) is selected as the optimal Euclidean point. This method ensures that the chosen point minimizes both the number of features and the algorithm's error simultaneously.

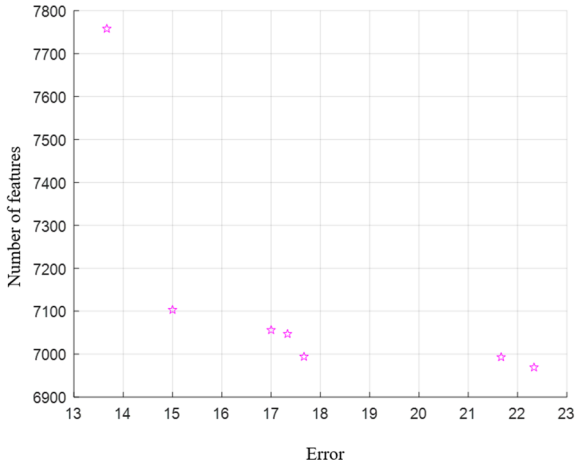

Fig. 16. Pareto front chart of the number of errors and features (ANN and GA).

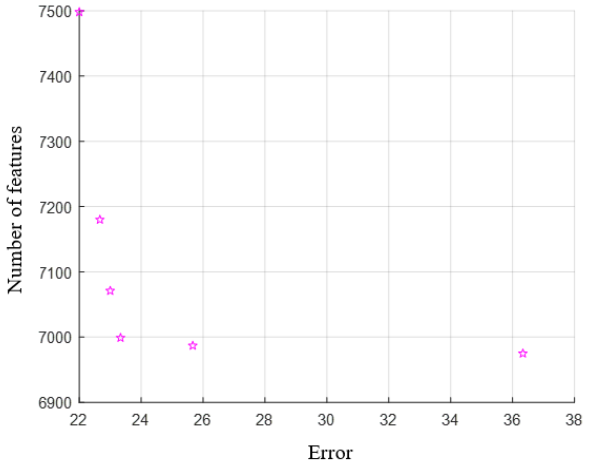

Fig. 17. Pareto front chart of the number of errors and features (SVM and GA).

Considering the results obtained from using GA and ANN (Fig. 16), each point on the chart specifies the average number of errors and the number of features used for detection. Considering the point with the least Euclidean distance from the origin (0,0), the optimal point was reported with 15 errors, and the number of features decreased from 16384 to 7103 after applying the dual-objective genetic algorithm optimization. The detailed values of the average number of features and their corresponding errors have been presented in Table 3. It is worth mentioning that, since this algorithm is executed across three sets of validation classes, the final error is the average error obtained from running the algorithm in three cross-validation sets.

Table 3. Number of features and their corresponding errors (ANN and GA)

Point	Average of error	Average number of features
1	13.66	7758
2	21.66	6993
3	17.33	7047
4	17.66	6994
5	15	7103
6	22.33	6969
7	17	7056

Similarly, considering the results obtained from using SVM and GA (Fig. 17), the optimal point was reported with an average of 23.33 errors, and the number of features was reduced from 16384 to 6999 after applying the dual-objective genetic algorithm optimization. The detailed values of the average number of features and their corresponding errors have been presented in Table 4.

Table 4. Number of features and their corresponding errors (SVM and GA)

Point	Average of error	Average number of features
1	22	7498
2	36.33	6975
3	23.13	6999
4	23	7071
5	22.67	7180
6	25.67	6987

F. Comparison of the Prediction Power of Different Algorithms

Following the implementation and evaluation of the proposed method using evaluation criteria derived from the confusion matrix, a comparison is conducted between existing methods. The overall accuracy of the different modeling approaches utilized in this study is shown in Fig. 18. The proposed method, which incorporates a feature selection approach utilizing the metaheuristic genetic algorithm optimization, not only identifies the best representative features related to Keratoconus within the dataset but also reduces classification errors in predicting test samples. This demonstrates a significant improvement compared to methods that did not use the GA. Moreover, the algorithm's predictive performance improved when employing artificial neural networks coupled with a genetic algorithm, surpassing that of SVM with a genetic algorithm for classification. In addition to accuracy, the superior performance of the algorithm combining artificial neural networks (ANN) with a genetic algorithm (GA) is further demonstrated by its higher precision, recall, and F1 score (Fig. 19).

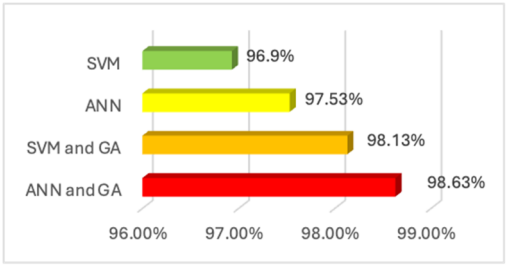

Fig. 18. Comparison of the modeling accuracy using different algorithms.

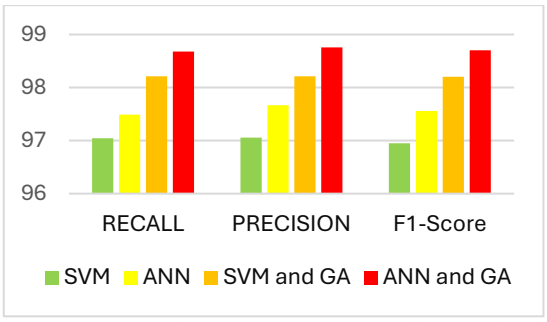

Fig. 19. Comparison of the performance metrics of different algorithms.

V. Conclusion

This study has clearly demonstrated that an integrated machine learning approach can be effectively applied to detect and classify Keratoconus (KCN). By combining Convolutional Neural Networks (CNN) for feature extraction with Support Vector Machines (SVM) and Artificial Neural Networks (ANN) for classification, significant advancements in the diagnosis of KCN have been achieved.

Reducing the size of the feature vector impacts classifier complexity by lowering the computational load and training time, reducing overfitting risks. In this study, before feature reduction, the classifier must process a vast number of features (16384), leading to high complexity, longer training times and increased overfitting risks. After feature reduction using a GA for SVM and ANN, the model becomes less complex, processing fewer features (SVM: 6,999 and ANN: 7,103), resulting in faster training and improved efficiency. This not only helped identify the most relevant features for Keratoconus detection in the dataset but also led to a significant reduction in classification errors on test samples compared to methods without GA.

It was found that the combination of CNN and GA for feature selection leads to more accurate diagnoses by effectively managing complex data, thus reducing the typical risks associated with handling large volumes of information. By focusing on the most relevant features of KCN, an accuracy rate of 98.63% was achieved by the model, which was higher than the 98.13% accuracy achieved by models utilizing SVM with GA. This underscores the advantages of integrating ANN with GA for this type of analysis. In addition to accuracy, the ANN and GA model significantly outperforms the other models across multiple evaluation metrics, including precision, recall, and F1 score. In other words, the ANN and GA model achieves the highest recall and precision rates, which lead to an improved F1 score, indicating a better balance between correctly identified positive instances and minimizing false positives.

The strength of the predictive models was also supported by the use of a large and varied database that included 1,288 real patients and 5,152 images. This extensive dataset played a critical role in improving the algorithm's capabilities, highlighting the importance of having both high-quality and large amounts of data to ensure high accuracy.

While this study is limited to a single dataset, the robustness of the approach, leveraging a combination of artificial neural networks and genetic algorithms, suggests it is likely to yield consistent results across different datasets, as these methods are well-established for their ability to generalize effectively across varying data distributions and feature spaces. Therefore, it is suggested that future studies apply this approach to different datasets to further validate its generalizability and effectiveness across diverse data distributions. To improve the prediction of Keratoconus, future research could also consider incorporating demographic factors such as gender and age, along with clinical measurements like vision and refraction, as these may influence the detection of asymptomatic Keratoconus. Additionally, exploring the use of curvelet transforms for image processing could be beneficial, especially for cases involving circular textures and edge details. Finally, investigating the integration of various classification methods with different feature subset selection techniques may enhance prediction accuracy by leveraging the strengths of multiple algorithms.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

This research paper is derived from the MSc thesis work conducted by Somayeh Ghasedi under the supervision of Amr R. Abdel-Dayem. The thesis was developed as part of the requirements for the Master of Science program in Computational Sciences at Laurentian University. The student was primarily responsible for the conceptualization, design, implementation, and analysis of the research, while the supervisor provided guidance, oversight, and critical feedback throughout the process.

This collaboration reflects the combined effort to translate the original thesis findings into a publishable research paper, ensuring broader dissemination and impact within the academic and professional community.

Acknowledgment

The authors thank Dr. Al-Timemy and Dr. Abdelmotaal for providing access to their topography images dataset.

References

[1] W. Wei, M. Ma, and Z. Liu, "Research progress of artificial intelligence in evaluating the efficacy of neoadjuvant chemotherapy for breast cancer," *EngMedicine,* vol. 1, no. 2, p. 100024, 2024.

[2] J. Liao, Y. Gui, Z. Li, Z. Deng, X. Han, H. Tian, L. Cai, X. Liu, C. Tang, J. Liu, Y. Wei, L. Hu, F. Niu, J. Liu, X. Yang, S. Li, X. Cui, X. Wu, Q. Chen, A. Wan, J. Jiang, Y. Zhang, X. Luo, P. Wang, Z. Cai, and L. Chen, "Artificial intelligence-assisted ultrasound image analysis to discriminate early breast cancer in Chinese population: A retrospective, multicentre, cohort study," *eClinicalMedicine,* vol. 60, pp. 102001, 2023/06/01/, 2023.

[3] X. Chen, Y.-L. A. Leung, and J. Shen, "Artificial intelligence and its application for cardiovascular diseases in Chinese medicine," *Digital Chinese Medicine,* vol. 5, no. 4, pp. 367–376, 2022.

[4] M. Wang, F. Francis, H. Kunz, X. Zhang, C. Wan, Y. Liu, P. Taylor, S. H. Wild, and H. Wu, "Artificial intelligence models for predicting cardiovascular diseases in people with type 2 diabetes: A systematic review," *Intelligence-Based Medicine,* vol. 6, p. 100072, 2022.

[5] Z. Yao, H. Wang, W. Yan, Z. Wang, W. Zhang, Z. Wang, and G. Zhang, "Artificial intelligence-based diagnosis of Alzheimer's disease with brain MRI images," *European Journal of Radiology,* vol. 165, p. 110934, 2023.

[6] C. A. Finney, F. Delerue, W. A. Gold, D. A. Brown, and A. Shvetcov, "Artificial intelligence-driven meta-analysis of brain gene expression identifies novel gene candidates and a role for mitochondria in Alzheimer's disease," *Computational and Structural Biotechnology Journal,* vol. 21, pp. 388–400, 2023.

[7] I. Bazarbekov, A. Razaque, M. Ipalakova, J. Yoo, Z. Assipova, and A. Almisreb, "A review of artificial intelligence methods for Alzheimer's disease diagnosis: Insights from neuroimaging to sensor data analysis," *Biomedical Signal Processing and Control,* vol. 92, p. 106023, 2024.

[8] A. Hussain, and A. Aslam, "Cardiovascular Disease Prediction Using Risk Factors: A comparative performance analysis of machine learning models," *Journal on Artificial Intelligence,* vol. 6, no. 1, pp. 129–152, 2024.

[9] H. Maile, J.-P. O. Li, D. Gore, M. Leucci, P. Mulholland, S. Hau, A. Szabo, I. Moghul, K. Balaskas, K. Fujinami, P. Hysi, A. Davidson, P. Liskova, A. Hardcastle, S. Tuft, and N. Pontikos, "Machine learning algorithms to detect subclinical keratoconus: systematic review," *JMIR Med Inform,* vol. 9, no. 12, p. e27363, 2021.

[10] N. Maeda, S. D. Klyce, M. K. Smolek, and H. W. Thompson, "Automated keratoconus screening with corneal topography analysis," *Invest Ophthalmol Vis Sci,* vol. 35, no. 6, pp. 2749–2757, May 1994.

[11] A. Lavric, L. Anchidin, V. Popa, A. H. Al-Timemy, Z. Alyasseri, H. Takahashi, S. Yousefi, and R. M. Hazarbassanov, "Keratoconus severity detection from elevation, topography and pachymetry raw data using a machine learning approach," *IEEE Access,* vol. 9, pp. 84344–84355, 2021.

[12] R. Feng, Z. Xu, X. Zheng, H. Hu, X. Jin, D. Z. Chen, K. Yao, and J. Wu, "KerNet: A novel deep learning approach for keratoconus and sub-clinical keratoconus detection based on raw data of the Pentacam HR System," *IEEE J Biomed Health Inform,* vol. 25, no. 10, pp. 3898–3910, Oct, 2021.

[13] A. H. Al-Timemy, N. H. Ghaeb, Z. M. Mosa, and J. Escudero, "Deep transfer learning for improved detection of keratoconus using corneal topographic maps," *Cognitive Computation,* vol. 14, no. 5, pp. 1627–1642, 2022.

[14] L. Balyen, and T. Peto, "Promising artificial intelligence-machine learning-deep learning algorithms in ophthalmology," *Asia Pac J Ophthalmol (Phila),* vol. 8, no. 3, pp. 264–272, 2019.

[15] K. Kamiya, Y. Ayatsuka, Y. Kato, N. Shoji, Y. Mori, and K. Miyata, "Diagnosability of keratoconus using deep learning with placido disk-based corneal topography," *Frontiers in Medicine,* vol. 8, 2021–October-04, 2021.

[16] A. Lavric, and P. Valentin, "KeratoDetect: Keratoconus detection algorithm using convolutional neural networks," *Computational Intelligence and Neuroscience,* vol. 2019, p. 8162567, 2019.

[17] O. Mehdizadeh Dastjerdi, M. Bakhtiarnia, M. Yazdchi, K. Maghooli, F. Farokhi, and K. Jadidi, "Ocular condition prognosis in Keratoconus patients after corneal ring implantation using artificial neural networks," *Heliyon,* vol. 9, no. 9, p. e19411, 2023.

[18] A. Elsawy, T. Eleiwa, C. Chase, E. Ozcan, M. Tolba, W. Feuer, M. Abdel-Mottaleb, and M. Abou Shousha, "Multidisease deep learning neural network for the diagnosis of corneal diseases," *American Journal of Ophthalmology,* vol. 226, pp. 252–261, 2021.

[19] A. H. Al-Timemy, L. Alzubaidi, Z. M. Mosa, H. Abdelmotaal, N. H. Ghaeb, A. Lavric, R. M. Hazarbassanov, H. Takahashi, Y. Gu, and S. Yousefi, "A deep feature fusion of improved suspected keratoconus detection with deep learning," *Diagnostics,* vol. 13, no. 10, p. 1689, 2023.

[20] S. P, and R. G. P, "Keratoconus Classification with convolutional neural networks using segmentation and index quantification of eye topography images by particle swarm optimisation," *BioMed Research International*, p. 8119685, 2022.

[21] M. C. Arbelaez, F. Versaci, G. Vestri, P. Barboni, and G. Savini, "Use of a support vector machine for keratoconus and subclinical keratoconus detection by topographic and tomographic data," *Ophthalmology,* vol. 119, no. 11, pp. 2231–2238, 2012.

[22] Y. Gao, Q. Wu, J. Li, J. Sun, and W. Wan, *SVM-Based Automatic Diagnosis Method for Keratoconus*, pp. WPR: SPIE, 2017.

[23] A. H. Al-Timemy, Z. M. Mosa, Z. Alyasseri, A. Lavric, M. M. Lui, R. M. Hazarbassanov, and S. Yousefi, "A hybrid deep learning construct for detecting keratoconus from corneal maps," *Translational Vision Science & Technology,* vol. 10, no. 14, p. 16, 2021.

[24] Mosa *et al*., "Detecting keratoconus by using SVM and decision tree classifiers with the aid of image processing," *Baghdad Science Journal,* vol. 16, no. 4(Suppl.), p. 1022, 2019.

[25] F. Toutounchian, J. Shanbezadeh, and M. Khanlari, "Detection of keratoconus and suspect keratoconus by machine vision," *Lecture Notes in Engineering and Computer Science,* vol. 2195, pp. 89–91, 2012.

[26] M. B. Souza, F. W. Medeiros, D. B. Souza, R. Garcia, and M. R. Alves, "Evaluation of machine learning classifiers in keratoconus detection from orbscan II examinations," *Clinics,* vol. 65, no. 12, pp. 1223–1228, 2010.

[27] M. S. del Río and J. P. Trevino, "Machine learning algorithms for classifying corneas by Zernike descriptors," *Intelligence-Based Medicine,* vol. 7, p. 100081, 2023.

[28] I. Kovács, K. Miháltz, K. Kránitz, É. Juhász, Á. Takács, L. Dienes, R. Gergely, and Z. Z. Nagy, "Accuracy of machine learning classifiers using bilateral data from a Scheimpflug camera for identifying eyes with preclinical signs of keratoconus," *Journal of Cataract & Refractive Surgery,* vol. 42, no. 2, 2016.

[29] M. Umar, F. Amin, H. A. Wahab, and D. Baleanu, "Unsupervised constrained neural network modeling of boundary value corneal model for eye surgery," *Applied Soft Computing,* vol. 85, p. 105826, 2019.

[30] Z. Sabir, M. Umar, H. A. Wahab, S. A. Bhat, and C. Unlu, "Heuristic computing performances based Gudermannian neural network to solve the eye surgery corneal model," *Applied Soft Computing,* vol. 157, p. 111540, 2024.

[31] K. He, X. Zhang, S. Ren, and J. Sun, *Deep Residual Learning for Image Recognition*, pp. 770–778.

[32] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, *Densely Connected Convolutional Networks*, pp. 2261–2269.

[33] B.-I. Kuo, W.-Y. Chang, T.-S. Liao, F.-Y. Liu, H.-Y. Liu, H.-S. Chu, W.-L. Chen, F.-R. Hu, J.-Y. Yen, and I.-J. Wang, "Keratoconus screening based on deep learning approach of corneal topography," *Translational Vision Science & Technology,* vol. 9, no. 2, pp. 53–53, 2020.

[34] P. Subramanian and G. P. Ramesh, "Keratoconus classification with convolutional neural networks using segmentation and index quantification of eye topography images by particle swarm optimisation," *Biomed Res Int*, pp. 8119685, 2022.

[35] Z. Li, H. Ma, D. Li, and R. Fan, "Genetic algorithm optimization of convolutional neural network for liver cancer CT image classification," in *Proc. 2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)* pp. 1075–1081.

[36] F. R. Llorella, G. Patow, and J. M. Azorín, "Convolutional neural networks and genetic algorithm for visual imagery classification," *Physical and Engineering Sciences in Medicine,* vol. 43, no. 3, pp. 973–983, 2020.

[37] A. Kabir Anaraki, M. Ayati, and F. Kazemi, "Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms," *Biocybernetics and Biomedical Engineering,* vol. 39, no. 1, pp. 63–74, 2019.

[38] K. Davoudi, and P. Thulasiraman, "Evolving convolutional neural network parameters through the genetic algorithm for the breast cancer classification problem," *SIMULATION,* vol. 97, no. 8, pp. 511–527, 2021.

[39] L. F. Rodrigues, A. R. Backes, B. A. N. Travençolo, and G. M. B. de Oliveira, "Optimizing a deep residual neural network with genetic algorithm for acute lymphoblastic leukemia classification," *J Digit Imaging,* vol. 35, no. 3, pp. 623–637, Jun, 2022.

[40] Z. Shah, S. Singh, N. Shilpy, and D. Purohit, "Prevalence of corneal topographic patterns in Ectatic corneal diseases," *Optom Vis Sci,* vol. 99, no. 7, pp. 560–567, Jul 1, 2022.

[41] K. Budak, A. M. Hamed, N. J. Friedman, and D. D. Koch, "Corneal topography classification in myopic eyes based on axial, instantaneous, refractive, and profile difference maps," *J Cataract Refract Surg,* vol. 25, no. 8, pp. 1069–79, Aug, 1999.

[42] M. Ming Wang and L. J. Kugler, *Atlas and Clinical Reference Guide for Corneal Topography*, SLACK Incorporated, 2013.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems,* vol. 25, 2012.