COVID-19 Variant Classification by Machine Learning, Signal Processing and Dimensionality Reduction

Love Fadia, Vatsal Shah, Mohammad Hassanzadeh, Majid Ahmadi, Jonathan Wu Department of Electrical and Computer Engineering
University of Windsor
Windsor, Canada
Email: fadial@uwindsor.ca, shah7r1@uwindsor.ca, mhassan@uwindsor.ca, M.ahmadi@uwindsor.ca, jwu@uwindsor.ca

Abstract—The far-reaching global impact of COVID-19 is unmistakable, accounting for an estimated loss of 1,110,000 lives attributed to the virus. The early identification of the virus and its various strains is imperative for safeguarding lives. Over the past few years multifarious machine learning and deep learning techniques were used to classify different genomic signals. However, not all the approaches that were used are as effective with different signal processing techniques. This article presents an efficient method for classifying coronavirus variants' DNA sequences using machine learning and signal processing. The DNA sequences are first converted into numbers using Electron-Ion Interaction Potential, Numeric, and Complex coding techniques after that signal processing methods; Discrete Cosine Transform-2, Discrete Cosine Transform-3, Fast Fourier Transform, Haar Wavelet Transform, and Coiflet Wavelet Transform are applied to extract features from the coded data. The high dimensionality is reduced using Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). For the classification task, machine learning models; Decision Tree, Support Vector Classifier, and a fusion of Light-Gradient Boosting Machine(LGBM), AdaBoost, and Random Forest are employed. The proposed approach achieves an impressive accuracy of 99.8% using a different combination of transformations with Numeric coding and Voting Classifier.

Index Terms—Genomic Sequence Analysis, Signal Processing, Dimensionality Reduction, Machine Learning.

1 Introduction

Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2), the virus that caused the COVID-19 pandemic, has evolved fast through various genetic alterations, resulting in the formation of several unique viral strains known as variants. These genetic variants in the SARS-CoV-2 genome have a significant impact on essential viral features such as transmissibility, immune response resistance, and the severity of infection. Understanding these alterations is critical for researchers to track the virus's progression and predict its impact on public health. Although SARS-CoV-2's genetic material is made up of RNA, researchers typically convert it to DNA for analysis, yielding stable DNA sequence data made up of four nitrogenous bases: adenine (A), thymine (T), cytosine (C), and guanine (G). This DNA format facilitates more reliable mutation tracking and offers insights into viral evolution in a consistent framework [1].

To date, five major SARS-CoV-2 variants—Alpha, Delta, Omicron, Beta, and Gamma—have been identified as having significant public health implications due to their enhanced transmissibility or increased potential for immune evasion [2]–[5]. Each of these variants carries unique genetic mutations that modify the virus's behavior, influencing factors such as the rate of viral spread, immune escape capabilities, and the overall severity of infection [6]. The variants Alpha, Delta, and Omicron, in particular, have been subject to extensive research and monitoring, given their notable effects on human health and their influence on public health strategies. Analyzing the genetic mutations within each of these variants allows researchers to trace the virus's evolutionary pathway and assess how its behavior changes over time [7], [8].

Traditionally, deep learning techniques have gained prominence in variant classification tasks due to their ability to handle large and complex genetic datasets effectively. Such models enable the automated identification and categorization of SARS-CoV-2 variants based on genetic sequence data, providing robust and scalable solutions [9]. However, deep learning models are often computationally intensive and require substantial processing power and memory resources, making them challenging to deploy in settings with limited computing capabilities. Even though pretrained models [10] offer some relief by reducing the need for extensive model training, they remain computationally demanding, which can limit their feasibility for large-scale or real-time applications.

To address these limitations, we present an efficient classification methodology that integrates signal processing techniques, dimensionality reduction, and machine learning classifiers to provide accurate variant classification with lower computational needs. To extract significant characteristics from DNA sequence data, we use five distinct signal processing techniques: discrete cosine transform-2 (DCT-2), discrete cosine transform-3 (DCT-3), fast Fourier transform (FFT), Haar wavelet, and coiflet wavelet. These techniques capture essential sequence patterns by converting DNA data to the frequency domain, where key features can be more easily discovered. We further simplify the dataset by using dimensionality reduction approaches, optimizing it for classification while retaining key variant-specific information.

The combination of signal processing and dimensionality reduction offers a more computationally efficient alternative to typical deep learning models, allowing for large-scale and real-time applications. Our technique allows for reliable categorization of SARS-CoV-2 variations while reducing processing time and memory utilization, which is especially useful in resource-constrained environments. This methodology maintains excellent classification accuracy by preserving essential properties associated with variant-specific mutations and leveraging them for robust identification.

The article is structured as follows: Section 2 covers related work, providing a review of relevant literature. Section 3 details the methodology, explaining each component in depth. Section 4 presents an analysis of the results from the proposed method. Finally, Section 5 offers conclusions and suggests areas for future research.

2 Related Work

This section describes various literature related to our work, which combines machine learning classifiers and signal processing techniques to advance SARS-CoV-2 virus classification. Khodel *et al.* [11] developed a method using Singular Value Decomposition, linear predictive feature extraction, and z-curve mapping to categorize coronavirus genomic signals with a 99% accuracy using Support Vector Machine (SVM). In related work, Naeem *et al.* [12] transformed DNA sequences into the frequency domain using Discrete Cosine and Fourier Transforms and applied a KNN model to achieve a high accuracy of 98.89%. Patel *et al.* [13] used wavelet transformation and statistical analysis to distinguish COVID-19-infected genes from normal genomes. Meng *et al.* [14] analyzed Wavelet Transform and Machine Learning applications for DNA sequences in cancer studies. Yadav *et al.* [15] mapped DNA sequences to complex numbers, applying the Short Time Ramanujan Fourier Transform and frequency-domain thresholding to extract sequence patterns accurately.

Chalco *et al.* [16] implemented a Modified Gabor Wavelet Transform with thresholding on coefficients for coding region identification. Randhawa *et al.* [17] achieved 98.1% accuracy in COVID-19 variant classification by combining supervised machine learning with digital signal processing using decision trees and Spearman's rank correlation. Muhammad *et al.* utilized eXtreme Gradient Boosting (XGB) and Light Gradient Boosting Machine (LGBM) with one-hot encoding, achieving 99.2% accuracy with XGB and faster computation with LGBM [18]. Hammad *et al.* [19] used Frequency Chaos representation to convert genomic signals into images, applying AlexNet for feature selection and reaching 99.71% accuracy with KNN and Decision Trees. Finally, Saha *et al.* [20] developed COVID-DeepPredictor, achieving up to 100% accuracy, while Eldosuky *et al.* [21] classified COVID-19 and influenza with 99% accuracy using optimized deep neural networks.Furthermore, datasets containing 66 HCoV sequences, 640 CoV-variant sequences, and 2,000 randomized SARS-CoV-2 sequences allowed for validation of wavelet basis effectiveness, with classification accuracies of up to 99%. These genomic signal processing algorithms, which include discrete wavelet decomposition with lifting, Fourier transform, and singular value decomposition, improve DNA sequence comparison by enabling alignment-free methods such as purine-pyrimidine mapping, DNA walks, and Z-curves. Such methodologies produced great classification accuracy across several Coronavirus datasets, with an ideal accuracy of 98.9% for CoV-Variants and up to 100% for certain HCoV strains, indicating a strong, cost-effective alternative for virus classification [22]. Moreover,Togrul.M *et al.* [23] used a deep learning model to pick features from DNA sequences for COVID-19 Variant classification, followed by a machine learning classifier for classification. SVM, KNN, MLP, and Random Forest are utilized for this purpose. Basu.S *et al.* [24] developed an alignment-free k-mer-based LSTM deep learning model to classify COVID-19 variations from genomic sequencing data. To resolve class imbalance, a set number of sequences are sampled for each class. The method addresses the vanishing gradient problem in LSTMs by breaking the sequences into fixed lengths and averaging the results over numerous runs.

The table 1 summarizes recent virus classification methods related to signal processing , Chaos game representation for feature and deep learning, machine learning classifier for classification, each with strengths and specific limitations. Khodel et al. used SVD, z-curve mapping, and SVM, achieving 99% accuracy on large datasets of coronavirus and influenza, but it demands high computational power for large datasets [11]. Naeem et al. used Discrete Cosine and Fourier Transforms with KNN, achieving 98.89% accuracy on a small COVID-related dataset, though accuracy may drop with more diverse data [12]. Randhawa et al.'s Decision Trees with Spearman's Rank achieved 98.1% accuracy, though it may not scale well with larger datasets [17]. Muhammad et al. used gradient boosting with one-hot encoding, reaching 99.2% accuracy, but one-hot encoding requires a lot of data and computing resources [18].

Hamaad M. et al. used FCGR and AlexNet, achieving 99.71%, but the method relies on pretrained models, which may not adapt well to new strains [19]. Saha et al.'s COVID-DeepPredictor reached nearly perfect accuracy but needs significant computing power [20]. Eldosuky et al.'s deep neural network achieved 99% but may be less effective across different virus types [21]. Kar et al. combined wavelet and Fourier methods, showing high accuracy on CoV-variant data, though it struggles with real-time processing [22]. The proposed method, with a voting classifier and advanced wavelets, achieved 99.8% accuracy and provides real-time efficiency but may still have limitations when handling highly diverse datasets.

3 METHODOLOGY

In our proposed work the multistage process commences with the initial collection of data, through the NCBI virus which is a specialized database within the NCBI infrastructure that focuses on viral genomic information [25]. We have taken 1000 DNA sequences each for 3 variants of the SARS-COV-2 virus in our study. Before applying any techniques to the data, we scan our data for any ambiguous sequences and remove them from the dataset so that the quality of our data remains supreme. After the cleaning stage, the DNA sequences undergo various coding techniques, which are Complex, EIIP, and Numeric. These coding methods play a pivotal role in transforming the raw data into more structured and analyzable forms. Furthermore, the coded representations are subjected to a transformation phase in the next step, where multifarious signal processing methods such as the Discrete Cosine Transform II, Discrete Cosine Transform III, Fast Fourier Transform, Haar Wavelet Transform, and Coiflet Wavelet Transform are employed. During this step of transformation, the data is transformed to the frequency domain, which aids in the capturing different patterns and characteristics from the coded representations. Following that, the dimensions of the frequency domain data are then reduced using Principal Component Analysis and Linear Discriminant Analysis, both techniques aimed at refining and identifying relevant characteristics that contribute to the accuracy of subsequent predictions. Our approach's last and most important step applies a variety of machine learning techniques. In this stage, we first seperate our dataset into training and testing sets. We have utilized 70% of our data for training and 30% for testing purposes. After the seperation of data machine learning techniques are applied. Following that, a thorough accuracy evaluation of the predictions is conducted to examine how well the different methodologies performed in capturing the complexity of different viral strains worked. This comparative study of prediction accuracies is an important evaluative step that will shed light on the general strength and effectiveness of our proposed work. Fig. 1 outlines the sequential stages of data processing, highlighting the collaborative role of coding, transformation, and machine learning for precise viral variant predictions. The table illustrates properties of dataset.

3.1 Coding Methods

3.1.1 EIIP coding

The method that converts DNA nucleotide sequences into numbers based on Electron Potential is known as EIIP coding. The method for these values' calculation is the atoms' potentials for electron-ion interactions [25]. The nucleotides of DNA have the following EIIP values:

- Adenine (A): 0.1260
- Thymine (T): 0.1335
- Guanine (G): 0.0806
- Cytosine (C): 0.1340

3.1.2 Complex coding

In this method, we assign 4 nucleotides of DNA sequence to complex numbers, which involves complementary characteristics [26]:

- Adenine (A): $1+i$
- Thymine (T): $-1-i$
- Guanine (G): $-1+i$
- Cytosine (C): $1-i$

3.1.3 Numeric coding

In Numeric coding, we transform DNA nucleotides into integers and it is a very fast and efficient approach. Here's how we can achieve it. [27].

- Adenine (A): 2
- Thymine (T): 0
- Guanine (G): 3
- Cytosine (C): 1

3.2 Signal Processing Methods

In our study, we use five linear transformations to take our data to the frequency domain.A key tool for compressing digital signals is the Discrete Cosine Transform II (DCT II). By concentrating signal energy in a small number of coefficients, particularly in the lower frequency components, it can express signals more succinctly. The equation yields the (DCT II). [28]:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right) \tag{1}$$

Here, X_k is the DCT coefficient at index k, x_n is the input sequence, N is the length of the sequence, and k ranges from 0 to $N-1$. The inverse of Discrete Cosine Transform II (DCT II) is Discrete Cosine Transform III (DCT III) given by the equation [29]

$$X_k = \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right) \tag{2}$$

.

An algorithmic method for quickly calculating the Discrete Fourier Transform (DFT) and its inverse is the Fast Fourier Transform (FFT). The DFT computation is substantially sped up by the FFT. The FFT significantly speeds up the computation of the DFT, The FFT is defined by the equation [30]

$$X[k] = FFT[x[n]] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \tag{3}$$

.

Haar Wavelet is a simple wavelet function that is popularly used in the field of Signal and Image processing due to piecewise linear functionality. Haar Wavelet equation is given by [31]:

$$\psi(x) = \begin{cases} 1, & \text{if } 0 \le x < \frac{1}{2}, \\ -1, & \text{if } \frac{1}{2} \le x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

where $\psi(x)$ represents the associated scaling function.

Coiflet wavelet is a mathematical function that is designed to have both properties; Vanishing Moments and Orthogonality. Vanishing Moments make sure that the wavelet follows a polynomial trend without affecting detailed coefficients. and Orthogonal property ensures that the inverse of this wavelet is proper. The equation is given as [32].

$$\psi(x) = \begin{cases} (1/\sqrt{2}) \times (\phi(x) - \phi(x-3)), & \text{if } 0 \le x < 3, \\ (1/\sqrt{2}) \times (-\phi(x-1) + \phi(x-4)), & \text{if } 3 \le x < 4, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

where $\psi(x)$ represents the associated scaling function.

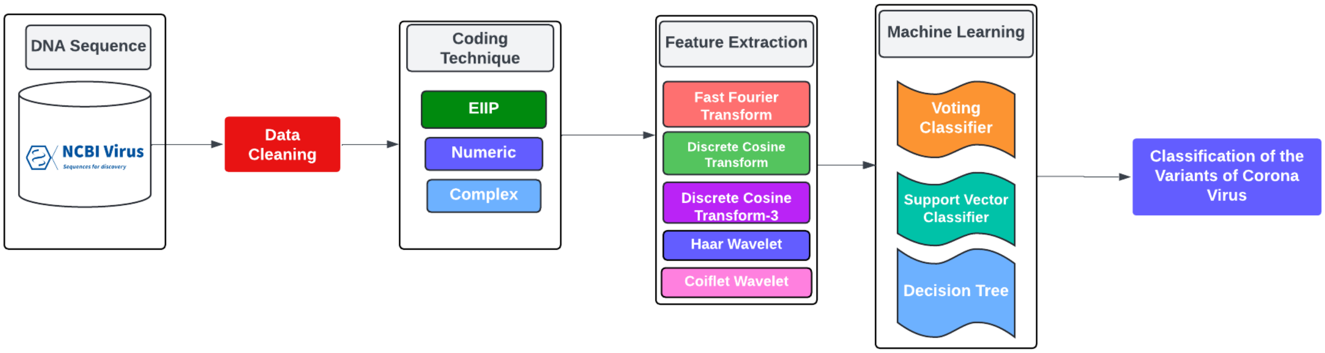

Fig. 1: Proposed Methodology

3.3 Dimensionality Reduction

Our original data has a maximum length of 30,000 base pairs and there are a total of 1000 DNA sequences for each variant. It can be seen that the size of the data is very large, i.e., $1000 \times 30,000$ columns for each variant class. Since there are a total of 3 different variants, resulting in 3 categories for classification. During machine learning operations, it was observed that computationally, it takes much more time to compute predictions due to such a huge size of data. To speed up the computation we use two dimensionality reduction techniques.

3.3.1 Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction technique that reduces the size of a dataset while preserving as much variance as possible. It basically transforms the raw data into a set of uncorrelated variables which are known as principal components, This is how it simplifies the complexity of dimension data [33]. It is observed that the first principal component has the highest amount of variations which goes on decreasing while moving further [34]. To calculate PCA, the eigenvalues and eigenvectors of the covariance matrix χ has been calculated. The eigenvalues ultimately represent the amount of variations explained by each principal component, while the eigenvectors represent the direction of these components. In our study, 60 principal components are used for classification.

3.3.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis is used to maximize the ratio of between-class variance to the within-class variance in any particular dataset, which leads to maximal separability among classes. The Scatter matrices play a vital role in this analysis which is defined as: [35].

$$S_W = \sum_{i=1}^{k} \Sigma(x - \mu_i)(x - \mu_i)^T \qquad (6)$$

Here, x belongs to the particular class i.

$$S_B = \sum_{i=1}^{k} N_i(\mu_i - \mu)(\mu_i - \mu)^T \qquad (7)$$

where N_i is the number of samples in class i, μ_i is the mean vector of class i, and μ is the overall mean vector of the dataset. The last step of LDA is to find a projection matrix W that maximizes the ratio of the determinant of the between-class scatter matrix to the determinant of the within-class scatter matrix [36]:

$$J(W) = \frac{|W^T S_B W|}{|W^T S_W W|} \qquad (8)$$

The optimal projection matrix W can be found by solving the generalized eigenvalue problem [35]:

$$S_W^{-1} S_B W = \Lambda W \qquad (9)$$

where Λ is a diagonal matrix whose entries are the eigenvalues. The eigenvectors corresponding to the largest eigenvalues form the columns of the projection matrix W.

3.4 Machine Learning Methods

3.4.1 Voting Classifier

A voting classifier is an ensemble method that combines the predictions of multiple machine learning models to produce a single, more accurate prediction by leveraging the strengths of diverse classifiers and mitigating their individual weaknesses [37]. We have used three models in our study to create a voting classifier, which are Light Gradient Boosting Machine(LGBM), Random Forest, and AdaBoost which can be seen in Fig.2

LGBM is a highly robust and efficient model that utilizes a technique called gradient boosting to construct an ensemble of decision trees. It combines the results or predictions of multiple weak learners to form a strong learner sequentially, and with each iteration, the trees learn from the errors of the previous ones [38]. The final prediction is the sum of the output of all the trees. Random Forest is a strong supervised ensemble learning approach that is frequently used for classification tasks. The working principle of Random Forest is to create a large number of decision trees and combine all of their predictions to increase accuracy and simultaneously decrease overfitting, and improve generalization. [39]. Adaboost short for Adaptive Boosting is also one of the best algorithms from the ensemble family. The main difference between the Adaboost and other decision tree algorithms is that; Adaboost uses stumps; which are nodes with two leaves to create the tree [40].

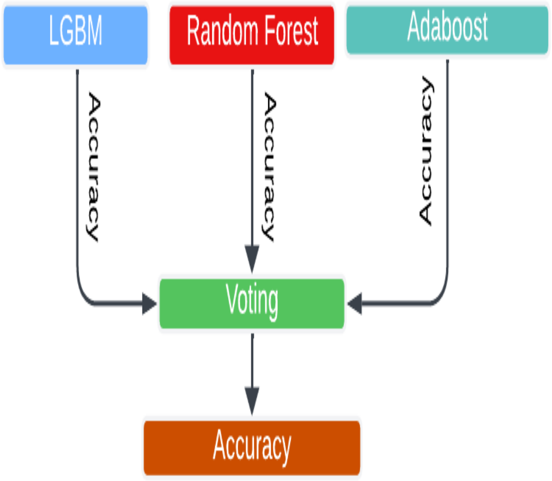

Fig. 2: Voting Classifier

3.4.2 *Decision Trees*

Decision Tree is a supervised learning machine learning method that uses a tree-like structure and nodes for options and leaves for outcomes, that produce decision rules based on data properties. Data is divided in a binary way at each node in accordance with predetermined rules in order to a construct decision tree. While building a decision tree a common practice is to utilise the CART (Classification and Regression Trees) algorithm. CART utilizes Gini impurity in classification issues to select the best node for splitting [41]. The number of times a randomly chosen element from the dataset would be incorrectly classified if it were labeled at random using the label distribution within that subset is measured by the Gini impurity which is defined as [42]:-

$$I_G(t) = 1 - \sum_{i=1}^{J} p_i^2 \tag{10}$$

where p_i is the probability of class i at node t and J is the number of classes.

3.4.3 *Support Vector Classifier*

Another kind of classification technique in machine learning, Support Vector Classifiers(SVCs) is well-known in the field for their efficiency when used for high-dimensional data. SVCs try to find the best hyperplane in the feature space to divide the classes. The hyperplane that maximizes the margin between the nearest data points of any class, or the support vector, is the optimal one [43]. Furthermore, in difficult classification tasks, SVCs are particularly useful due to their ability to maximize margin on top of that, they can also perform non-linear classification effectively with the use of kernel strategies [44].

4 Experimental Results

4.1 Results comparision for PCA

Table 1 summarizes the study results after applying Principal Component Analysis (PCA) for dimensionality reduction. The accuracy of clssification is greatly impacted by the selection of machine learning algorithm, transformation, and coding method.With the FFT transformation, the Voting Classifier gets the best accuracy for the EIIP coding approach (98.65%), followed by DCT-II (96.2%). The Voting Classifier Haar Wavelet displays a lower accuracy of 86.86%. With EIIP coding, Decision Tree and SVC exhibit lower accuracy rates; Decision Tree ranges from 61% to 88%, while SVC ranges from 50% to 94%. The Voting Classifier and Haar Wavelet transformation together produce the maximum accuracy, 99.21%, when used with the Numeric Coding method. While Coiflet Wavelet attains 95.57% accuracy, DCT-II and DCT-III consistently display accuracy levels around 95.95%. Lower accuracy is shown by Decision Tree and SVC with Numeric coding; Decision Tree ranges from 65% to 77%, and SVC from 62% to 93%. Depending on the machine learning algorithm and transformation employed, the accuracy of the complex coding technique varies signif-

TABLE 1: Experimental Results for Principal Component Analysis

Sr No	Coding Techniques	Transformation	ML Techniques	Accuracy
1	EIIP	DCT-II	Voting Classifier	96.2%
2	EIIP	DCT-III	Voting Classifier	95.8%
3	EIIP	FFT	Voting Classifier	98.65%
4	EIIP	Haar Wavelet	Voting Classifier	86.86%
5	EIIP	Coiflet Wavelet	Voting Classifier	95.06%
6	Numeric	DCT-II	Voting Classifier	95.95%
7	Numeric	DCT-III	Voting Classifier	95.95%
8	Numeric	FFT	Voting Classifier	98.31%
9	Numeric	Haar Wavelet	Voting Classifier	99.21%
10	Numeric	Coiflet Wavelet	Voting Classifier	95.7%
11	Complex	DCT-II	Voting Classifier	96.18%
12	Complex	DCT-III	Voting Classifier	97.19%
13	Complex	FFT	Voting Classifier	97.5%
14	Complex	Haar Wavelet	Voting Classifier	80%
15	Complex	Coiflet Wavelet	Voting Classifier	80%
16	EIIP	DCT-II	Decision Tree	64%
17	EIIP	DCT-II	SVC	56%
18	Numeric	DCT-II	Decision Tree	67%
19	Numeric	DCT-II	SVC	64%
20	Complex	DCT-II	Decision Tree	72%
21	Complex	DCT-II	SVC	70%
22	EIIP	DCT-III	Decision Tree	75%
23	EIIP	DCT-III	SVC	89%
24	Numeric	DCT-III	SVC	62%
25	Numeric	DCT-III	Decision Tree	59%
26	Complex	DCT-III	SVC	71%
27	Complex	DCT-III	Decision Tree	70%
28	EIIP	FFT	Decision Tree	88%
29	EIIP	FFT	SVC	94%
30	Numeric	FFT	Decision Tree	77%
31	Numeric	FFT	SVC	93%
32	Complex	FFT	Decision Tree	81%
33	Complex	FFT	SVC	95%
34	EIIP	Haar Wavelet	Decision Tree	61%
35	EIIP	Haar Wavelet	SVC	50%
36	Numeric	Haar Wavelet	Decision Tree	86%
37	Numeric	Haar Wavelet	SVC	84%
38	Complex	Haar Wavelet	Decision Tree	57%
39	Complex	Haar Wavelet	SVC	50%
40	EIIP	Coiflet Wavelet	Decision Tree	65%
41	EIIP	Coiflet Wavelet	SVC	52%
42	Numeric	Coiflet Wavelet	Decision Tree	65%
43	Numeric	Coiflet Wavelet	SVC	63%
44	Complex	Coiflet Wavelet	Decision Tree	62%
45	Complex	Coiflet Wavelet	SVC	67%

icantly. For complex coding, the Voting Classifier FFT transformation yields the highest accuracy of 97.5%, followed by DCT-III at 97.19%. The accuracy of the Haar Wavelet transformation is the lowest, at 80%. With application complex coding, the accuracy of Decision Tree and SVC is similarly reduced; Decision Tree ranges from 57% to 81%, while SVC ranges from 50% to 95%. In general, the investigation shows that is a lot of variations in acuuracies in each of the 45 cases presented here.

4.1.1 Performance Analysis of Signal Processing and Machine Learning Models with EIIP coding for PCA

The accuracy of three distinct machine learning algorithms—Voting Classifier, Decision Tree, and SVC—when paired with EIIP coding and diverse signal processing transformations is assessed in this section. The Voting Classifier obtains a high accuracy of 96.2% with DCT-II transformation, while the accuracies of Decision Tree and SVC are much lower, at 64% and 56%, respectively. The Voting Classifier continues to perform well for DCT-III, recording 95.8%; SVC follows it closely with at 89%, while Decision Tree records 75%. The Voting Classifier gives the best accuracies (98.65%, SVC 94%, and Decision Tree 88%) when the FFT transformation is used, demonstrating the efficacy of FFT. Besides this, all classifiers, including the Voting Classifier (86.86%), Decision Tree (61%), and SVC (50%), have much worse accuracy when the Haar Wavelet transformation is applied. The Voting Classifier finally achieves 95.06% accuracy after applying Coiflet Wavelet transformation, outperforming Decision Tree and SVC at 65% and 52%, respectively, proving its supremacy.

4.1.2 Performance Analysis of Signal Processing and Machine Learning Models with Numeric coding for PCA

The accuracy of three machine learning algorithms—the Voting Classifier, Decision Tree, and SVC utilizing Numeric coding with different signal processing transformations—is compared in this section. The Voting Classifier attains 95.95% accuracy for the DCT-II transformation, while Decision Tree and SVC fall short at 67% and 64%, respectively. The Voting Classifier remains at 95.95% with DCT-III, while SVC falls to 62% and Decision Tree to 59%. The Voting Classifier produces the highest accuracy at 98.31%, SVC at 93%, and Decision Tree at 77% when using the FFT transformation. By comparison, the Voting Classifier achieves the maximum accuracy of 99.21% when using the Haar Wavelet transformation, while Decision Tree and SVC also demonstrate good performance, at 86% and 84%, respectively. Lastly, with Coiflet Wavelet transformation, the Voting Classifier achieves 95.7%, while Decision Tree and SVC perform moderately at 65% and 63%, respectively.

4.1.3 Performance Analysis of Signal Processing and Machine Learning Models with Complex coding for PCA

The accuracy of the Voting Classifier, Decision Tree, and SVC employing complex coding with different signal processing transformations is compared in this section. Voting Classifier scores 96.18% for DCT-II, whilst Decision Tree and SVC perform worse at 72% and 70%, respectively. The Voting Classifier achieves 97.19% with DCT-III, way ahead of SVC at 71% and Decision Tree at 70%. The Voting Classifier produces 97.5% accuracy, SVC 95%, and Decision Tree 81%; these results demonstrate the efficacy of the FFT transformation. On the other hand, accuracy is greatly decreased by Haar Wavelet transformation, with the Voting Classifier at 80%, Decision Tree at 57%, and SVC at 50%. The Voting Classifier, at 80%, is the last product of the Coiflet Wavelet transformation; Decision Tree and SVC, at 62% and 67%, respectively, perform moderately.

TABLE 2: Experimental Results for Linear Discriminant Analysis

Sr No	Coding Techniques	Transformation	ML Techniques	Accuracy
1	EIIP	DCT-II	Voting Classifier	99.6%
2	EIIP	DCT-III	Voting Classifier	99.8%
3	EIIP	FFT	Voting Classifier	99.7%
4	EIIP	Haar Wavelet	Voting Classifier	99.5%
5	EIIP	Coiflet Wavelet	Voting Classifier	99.6%
6	Numeric	DCT-II	Voting Classifier	99.8%
7	Numeric	DCT-III	Voting Classifier	99.4%
8	Numeric	FFT	Voting Classifier	99.8%
9	Numeric	Haar Wavelet	Voting Classifier	99.8%
10	Numeric	Coiflet Wavelet	Voting Classifier	99.3%
11	Complex	DCT-II	Voting Classifier	99.7%
12	Complex	DCT-III	Voting Classifier	99.3%
13	Complex	FFT	Voting Classifier	99.6%
14	Complex	Haar Wavelet	Voting Classifier	99.2%
15	Complex	Coiflet Wavelet	Voting Classifier	99.4%
16	EIIP	DCT-II	Decision Tree	99%
17	EIIP	DCT-II	SVC	97.3%
18	Numeric	DCT-II	Decision Tree	99%
19	Numeric	DCT-II	SVC	98.65%
20	Complex	DCT-II	Decision Tree	99%
21	Complex	DCT-II	SVC	98.53%
22	EIIP	DCT-III	Decision Tree	99%
23	EIIP	DCT-III	SVC	97.52%
24	Numeric	DCT-III	SVC	98.03%
25	Numeric	DCT-III	Decision Tree	99%
26	Complex	DCT-III	SVC	98.05%
27	Complex	DCT-III	Decision Tree	99%
28	EIIP	FFT	Decision Tree	99%
29	EIIP	FFT	SVC	98.08%
30	Numeric	FFT	Decision Tree	99%
31	Numeric	FFT	SVC	97.97%
32	Complex	FFT	Decision Tree	99%
33	Complex	FFT	SVC	97.97%
34	EIIP	Haar Wavelet	Decision Tree	99%
35	EIIP	Haar Wavelet	SVC	97.86%
36	Numeric	Haar Wavelet	Decision Tree	99%
37	Numeric	Haar Wavelet	SVC	98.42%
38	Complex	Haar Wavelet	Decision Tree	99%
39	Complex	Haar Wavelet	SVC	96.74%
40	EIIP	Coiflet Wavelet	Decision Tree	99%
41	EIIP	Coiflet Wavelet	SVC	97.86%
42	Numeric	Coiflet Wavelet	Decision Tree	99%
43	Numeric	Coiflet Wavelet	SVC	98.65%
44	Complex	Coiflet Wavelet	Decision Tree	99%
45	Complex	Coiflet Wavelet	SVC	97.19%

4.2 Results comparision for LDA

Table 2 summarizes the study results after applying Linear Discriminant Analysis (LDA) for dimensionality reduction. It can be confirmed here as well similar to PCA that the accuracy of classification is significantly affected by the selection of machine learning algorithm, transformation, and coding method. High accuracy seems to be achieved by combining different transformations and Voting Classifier with the EIIP coding technique; all transformations produce results between 99.5% and 99.8%. In particular, the Voting Classifier in conjunction with the DCT-III transformation yields the greatest accuracy of 99.8%. The accuracy is significantly reduced when utilizing Decision Tree and SVC with EIIP coding, particularly with SVC, ranging from 97.3% to 98.65%. Furthermore, the Voting Classifier continuously achieves high accuracy using the FFT transformation: 99.7% for EIIP, 99.8% for Numeric, and 99.6% for Complex coding approaches. With the Voting Classifier, the Haar Wavelet transformation achieves up to 99.8% accuracy, demonstrating competitive accuracy, especially when used with the Numeric coding approach. Besides that, LDA, the Voting Classifier and the DCT-II and DCT-III transformations consistently show good accuracy across a variety of coding strategies. The Voting Classifier's efficacy is illustrated by the DCT-II transformation, which provides 99.6% for EIIP, 99.8% for Numeric, and 99.7% for Complex coding. After LDA, Decision Tree and SVC typically perform less accurately than the Voting Classifier; nevertheless, Decision Tree consistently achieves 99% accuracy across a range of transformations and coding strategies. On the other hand, SVC exhibits a greater range of accuracy, varying from 96.74% to 98.65% contingent upon the coding method and transformation applied. This indicates that the Voting Classifier performs well in a variety of settings, but the Decision Tree and SVC classifiers are impacted by the transformation and coding method selected. In summary, the analysis highlights how crucial it is to choose the right mixes of coding strategies, transformations, and machine learning algorithms in order to attain the best classification results.

4.2.1 Performance Analysis of Signal Processing and Machine Learning Models with EIIP coding for LDA

Using EIIP coding with different signal processing transformations, the accuracy of three machine learning algorithms—Voting Classifier, Decision Tree, and SVC—is assessed in this section. 99.6% with DCT-II, 99.8% with DCT-III, 99.7% with FFT, 99.5% with Haar Wavelet, and 99.6% with Coiflet Wavelet are the highest performance levels regularly achieved by the Voting Classifier. The robustness of the Decision Tree in handling EIIP encoded data is demonstrated by its strong performance, which maintains 99% accuracy across all transformations. Although the SVC algorithm performs better with FFT (98.08%) and DCT-II (97.3%), it performs worse with Haar and Coiflet Wavelets (97.86%). In comparison, the SVC algorithm exhibits greater variability. Comparing SVC to the other models, this variability indicates that SVC has difficulty capturing the complexity of EIIP transformations to the fullest. Though not as effectively as the Voting Classifier and Decision Tree, SVC still shows some promise in spite of these difficulties.

4.2.2 Performance Analysis of Signal Processing and Machine Learning Models with Numeric coding for LDA

In this part, the accuracy of three machine learning algorithms—SVC, Voting Classifier, and Decision Tree—is compared using different transformations and Numeric coding. For Numeric DCT-II, the Voting Classifier has the highest accuracy at 99.8%, followed by Decision Tree at 99% and SVC at 98.65%. The Voting Classifier outperforms the Decision Tree at 99% and SVC at 98.03% for Numeric DCT-III, however it decreases somewhat to 99.4%. The Voting Classifier and Decision Tree with Numeric FFT attain 99.8% and 99%, respectively, whereas SVC achieves 97.97%, indicating the effectiveness of FFT in enhancing classification. The Voting Classifier retains 99.8%, the Decision Tree reaches 99%, and the SVC increases to 98.42% in the Numeric Haar Wavelet transformation. Last but not least, the Voting Classifier records 99.3%, Decision Tree keeps 99%, and SVC records 98.65% with Numeric Coiflet Wavelet, demonstrating a slight decline in performance for all models.

4.2.3 Performance Analysis of Signal Processing and Machine Learning Models with Complex coding for LDA

This section examines the accuracy of the Decision Tree, SVC, and Voting Classifier using Complex coding with various transformations. The Voting Classifier does better than the Decision Tree (99%) and SVC (98.53%) with a 99.7% accuracy rate for DCT-II, exhibiting good model performance with little variations. While the Decision Tree maintains its 99% level and the SVC achieves 98.05%, the Voting Classifier for DCT-III slightly declines to 99.3%. The voting classifier, decision tree, and SVC all perform exceptionally well in the complex FFT setup, with the voting classifier at 99.6%, decision tree at 99%, and SVC at 97.97%. This shows how powerful FFT is in enhancing classification accuracy.. After applying Haar Wavelet transformation, the Voting Classifier achieves 99.2%, the Decision Tree maintains its 99%, and SVC falls to 96.74%. Finally, the Voting Classifier scores 99.4% in the Coiflet Wavelet transformation, Decision Tree keeps scoring 99%, and SVC scores 97.19%. These results show some difficulties with complex feature extraction but overall good performance.

TABLE 3: Coding Techniques Analysis for PCA and LDA

Technique	Dimensionality Reduction	Avg. Acc.	Std Dev	Worst Acc.	Best Acc.
EIIP	PCA	75.74	0.1777	50	98.65
Numeric	PCA	80.34	0.155651	59	99.21
Complex	PCA	76.39	0.150266	50	97.5
EIIP	LDA	98.78	0.008431	97.3	99.8
Numeric	LDA	99.01	0.005704	97.97	99.8
Complex	LDA	98.71	0.008629	96.74	99.7

4.3 Comparative analysis of Dimensionality Reduction Techniques

Table 3 examines three coding techniques—EIIP, numeric, and complex for their average accuracy, standard deviation, worst accuracy, and highest accuracy across Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). The Numeric method under PCA has the best average accuracy (80.34%) and the lowest standard deviation (0.155651), with maximum and worst accuracy of 99.21% and 59%. EIIP has the lowest average accuracy (75.74%) and the highest variability, with values ranging from 50% to 98.65%. The Complex technique has a comparable, but more consistent, average accuracy of 76.39%. All methods improve when combined with LDA. With less variability and an accuracy range of 97.3% to 99.8%, the average accuracy of EIIP rises to 98.78% (0.008431). With the lowest standard deviation (0.005704) and the highest average accuracy (99.01%), the numeric coding technique performs consistently between 97.97% and 99.8%. With a complex approach, moderate variability and accuracy of 98.71% are achieved (0.008629). For all coding approaches combined, LDA greatly improves accuracy and consistency; Numeric performs particularly well in this regard.

TABLE 4: Signal Processing Techniques Analysis for PCA and LDA

Technique	Dimensionality Reduction	Avg. Acc.	Std Dev	Worst Acc.	Best Acc.
DCT-II	PCA	75.7	0.1594	56	96.2
DCT-III	PCA	79.44	0.152	59	97.19
FFT	PCA	89.35	0.078013	77	98.65
Haar Wavelet	PCA	72.67	0.1826	50	99.21
Coiflet Wavelet	PCA	71.64	0.15249	52	95.7
DCT-II	LDA	98.75	0.007668	97	99.8
DCT-III	LDA	98.79	0.0061911	97.53	99.8
FFT	LDA	98.90	0.00738	97.97	99.8
Haar Wavelet	LDA	98.72	0.009345	96.74	99.8
Coiflet Wavelet	LDA	98.85	0.00626	97.86	99.6

Table 4 provides a study of several signal processing methods under PCA and LDA, including DCT-II, DCT-III, FFT, Haar Wavelet, and Coiflet Wavelet. The average accuracy, standard deviation of accuracy, worst accuracy, and best accuracy are used to evaluate each technique. With accuracy ranging from 77% to 98.65%, FFT has the highest average accuracy of 89.35% and the lowest standard deviation of 0.078 under PCA, illustrating consistent performance. DCT-III comes next, with a 79.44% average accuracy, a 0.152 standard deviation, and accuracy that ranges from 59% to 97.19%. DCT-II displays an accuracy range of 56% to 96.2% and an average of 75.7% and a standard deviation of 0.159. With a wide accuracy range from 50% to 99.21% and considerable variability (standard deviation of 0.183), the Haar Wavelet approach has a lower average accuracy of 72.67%. With an accuracy range of 52% to 95.7%, an accuracy average of 71.64%, and a standard deviation of 0.152, Coiflet Wavelet likewise performs worse under PCA.All methods considerably improve under LDA. With a low standard deviation of 0.007 and an average accuracy of 98.90%, FFT continues to perform well, with accuracy ranging from 97.97% to 99.8%. With average accuracy of 98.75% and 98.79%, respectively, and narrow accuracy ranges, DCT-II and DCT-III likewise perform admirably under LDA. The Haar Wavelet increases from 96.74% to 99.8% in accuracy, with an average accuracy of 98.72% and a standard deviation of 0.009. The Coiflet Wavelet has an accuracy range of 97.86% to 99.6%, an average of 98.85%, and a standard deviation of 0.006.When compared to PCA, LDA enhances performance across the board, yielding better average accuracies and lower variability.

TABLE 5: ML Techniques Analysis for PCA and LDA

Technique	Dimensionality Reduction	Avg. Acc.	Std Dev	Worst Acc.	Best Acc.
Voting Classifier	PCA	93.9	0.06307	80	99.21
Decision Tree	PCA	70	0.096693	57	88
SVC	PCA	71	0.163867	50	95
Voting Classifier	LDA	99.57	0.002059	99.3	99.81
Decision Tree	LDA	99	0	99	99
SVC	LDA	97.93	0.00533	97.19	98.65

Table 5 presents an overview of the various machine learning algorithms' performances under PCA and LDA, including Voting Classifier, Decision Tree, and SVC. The Voting Classifier, under PCA, has the maximum average accuracy of 93.9%, ranging from 80% to 99.21%, with a standard deviation of 0.06307. With greater variability, Decision Tree and SVC exhibit lower average accuracies of 70% and 71%, respectively. SVC is between 50% and 95%, and Decision Tree is between 57% and 88%. All strategies demonstrate a notable improvement under LDA. With an extremely low standard deviation of 0.002059, the Voting Classifier achieves an average accuracy of 99.57%, ranging from 99.3% to 99.81%. With no fluctuation, Decision Tree continuously maintains 99% accuracy. With a standard deviation of 0.00533, SVC increases to an average accuracy of 97.93%, ranging from 97.19% to 98.6%. Overall, it can be seen that LDA enhances the accuracy and consistency of all algorithms, with the Voting Classifier performing the best.

TABLE 6: Comparison of Dimensionality Reduction Techniques

Technique	Avg. Acc.	Std Dev	Worst Acc.	Best Acc.
PCA	78%	0.1577	52%	99.21%
LDA	96.74%	0.0075	96.74%	99.80%

A thorough comparison of Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) with an emphasis on important performance indicators is provided in Table 6. The accuracy achieved using PCA is as follows: 52% at lowest, 99.21% at maximum, 78.17% at mean, and 0.1577 at standard deviation. With a mean accuracy of 96.74%, a maximum accuracy of 99.80%, a minimum accuracy of 96.74%, and a substantially smaller standard deviation of 0.00755, LDA, on the other hand, performs noticeably better than PCA. The better performance and consistency of LDA are demonstrated by this data. LDA is the best option for data processing tasks needing great precision and stability, as Table VI amply illustrates its benefits. LDA assures higher dependability by reducing variability and improving average accuracies in complicated data contexts where precision and reliability are critical. This essentially makes LDA a very important tool for multifarious data analysis projects, offering improved performance and uniformity compared to PCA.

5 Conclusion and Future Work

In summary, this study looked at how different data coding techniques and transformations interact to improve the performance of machine learning models, with a particular emphasis on PCA and LDA. We assessed EIIP, Numeric, and Complex coding schemes using FFT, DCT-II, DCT-III, Haar, and Coiflet wavelets, among other transformations. Our results show that when FFT is applied with Voting Classifier in case of PCA, it performs much better than other transformations for EIIP and Numeric data codings. Due to its improved performance, FFT is the best option for complicated and noisy datasets since it can extract important frequency-domain information. On the other hand, the performance of the Haar and Coiflet wavelet transformations varied, indicating that the context may affect how successful they are. It's possible that these wavelets don't reliably capture the required characteristics for various coding schemes. LDA also performed exceptionally well with all coding schemes and transformations, reliably generating a discriminative feature space that improves class separability with a high degree of accuracy. This study emphasizes how crucial it is to use the right coding schemes and transformations depending on the features of the dataset, helping experts to maximize machine learning model performance and produce superior classification outcomes. For future work, we plan to incorporate a more diverse dataset, including additional viral genomes, to broaden the classification scope and further validate the effectiveness of these techniques across varied viral data. This expansion will allow us to examine the generalizability of our approach and explore its potential in a wider range of applications.

References

[1] "Coronaviruses," NIH: National Institute of Allergy and Infectious Diseases, Mar. 22, 2022. Available: https://www.niaid.nih.gov/diseases-conditions/coronaviruses.

[2] J. Emrani., "SARS-COV-2, infection, transmission, transcription, translation, proteins, and treatment: A review," *International Journal of Biological Macromolecules*, vol. 193, pp. 1249–1273, Dec. 2021, doi: 10.1016/j.ijbiomac.2021.10.172.

[3] K. Rosenke., "UK B.1.1.7 (Alpha) variant exhibits increased respiratory replication and shedding in nonhuman primates," *Emerging Microbes and Infections*, vol. 10, no. 1, pp. 2173–2182, Jan. 2021, doi: 10.1080/22221751.2021.1997074.

[4] M. Dhawan, A. Sharma, P. Choudhary, N. Thakur, T. K. Rajkhowa, and O. P. Choudhary, "Delta variant (B.1.617.2) of SARS-CoV-2: Mutations, impact, challenges and possible solutions," *Human Vaccines and Immunotherapeutics*, vol. 18, no. 5, May 2022, doi: 10.1080/21645515.2022.2068883.

[5] R. Viana., "Rapid epidemic expansion of the SARS-CoV-2 Omicron variant in southern Africa," *Nature*, vol. 603, no. 7902, pp. 679–686, Jan. 2022, doi: 10.1038/s41586-022-04411-y.

[6] Cosar B, Karagulleoglu ZY, Unal S, Ince AT, Uncuoglu DB, Tuncer G, Kilinc BR, Ozkan YE, Ozkoc HC, Demir IN, Eker A, Karagoz F, Simsek SY, Yasar B, Pala M, Demir A, Atak IN, Mendi AH, Bengi VU, Cengiz Seval G, Gunes Altuntas E, Kilic P, Demir-Dora D. "SARS-CoV-2 Mutations and their Viral Variants. Cytokine Growth Factor Rev". 2022 Feb;63:10-22. doi:10.1016/j.cytogfr.2021.06.001.

[7] S. Amin, A. Alharbi, M. I. Uddin, and H. Alyami, "Adapting recurrent neural networks for classifying public discourse on COVID-19 symptoms in Twitter content," *Soft Computing*, vol. 26, no. 20, pp. 11077–11089, Aug. 2022, doi: 10.1007/s00500-022-07405-0.

[8] A. Aleem, A. B. A. Samad, and S. Vaqar, "Emerging variants of SARS-COV-2 and novel therapeutics against coronavirus (COVID-19)," *StatPearls - NCBI Bookshelf*, May 08, 2023. https://www.ncbi.nlm.nih.gov/books/NBK570580/

[9] S. Agarwal, K. V. Arya, and Y. K. Meena, "MultiFusionNet: multilayer multimodal fusion of deep neural networks for chest X-ray image classification," *Soft Computing*, Jul. 2024, doi: 10.1007/s00500-024-09901-x.

[10] W. Hariri and A. Narin, "Deep neural networks for COVID-19 detection and diagnosis using images and acoustic-based techniques: a recent review," *Soft Computing*, vol. 25, no. 24, pp. 15345–15362, Aug. 2021, doi: 10.1007/s00500-021-06137-x.

[11] A. Khodaei, P. Shams, H. Sharifi, and B. M. Tazehkand, "Identification and classification of coronavirus genomic signals based on linear predictive coding and machine learning methods," *Biomedical Signal Processing and Control*, vol. 80, p. 104192, Feb. 2023, doi: 10.1016/j.bspc.2022.104192.

[12] S. M. Naeem, M. S. Mabrouk, S. Y. Marzouk, and M. A. Eldosoky, "A diagnostic genomic signal processing (GSP)-based system for automatic feature analysis and detection of COVID-19," *Briefings in Bioinformatics*, vol. 22, no. 2, pp. 1197–1205, Aug. 2020, doi: 10.1093/bib/bbaa170.

[13] K. Patel, V. Shah, N. Patel and Y. Mehta, "An Non- Invasive Approach of Corona Genome Detection," 2020 International Conference on Advances in Computing, Communication / Materials (ICACCM), Dehradun, India, 2020, pp. 154-157, doi: 10.1109/ICACCM50413.2020.9213053.

[14] T. Meng .,Soliman A.,Shyu.M.,Yang.Y.,Chen.S.,lyengar.S, "Wavelet Analysis in Current Cancer Genome Research: A Survey," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 6, pp. 1442-14359, Nov.-Dec. 2013, doi: 10.1109/TCBB.2013.134.

[15] Y. Yadav, S. N. Sharma and D. K. Shakya, "Detection of Tandem Repeats in DNA Sequences Using Short-Time Ramanujan Fourier Transform," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 19, no. 3, pp. 1583-1591, 1 May-June 2022, doi:10.1109/TCBB.2021.3053656.

[16] J. Mena-Chalco, H. Carrer, Y. Zana and R. M. Cesar Jr., "Identification of Protein Coding Regions Using the Modified Gabor-Wavelet Transform," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 5, no. 2, pp. 198-207, April-June 2008, doi: 10.1109/TCBB.2007.70259.

[17] G. S. Randhawa, M. P. M. Soltysiak, H. E. Roz, C. P. E. De Souza, K. A. Hill, and L. Kari, "Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: COVID-19 case study," *PLOS ONE*, vol. 15, no. 4, p. e0232391, Apr. 2020, doi: 10.1371/journal.pone.0232391.

[18] I. Muhammad, I. Mukhlash, M. Jamhuri, M. Iqbal and M. I. Irawan, "Classification of Covid-19 Variants Using Boosting Algorithm," 2022 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Jakarta, Indonesia, 2022, pp. 29-34, doi: 10.23919/EECSI56542.2022.9946452.

[19] M. S. Hammad, V. F. Ghoneim, M. S. Mabrouk, and W. Al- Atabany, "A hybrid deep learning approach for COVID-19 detection based on genomic image processing techniques," *Scientific Reports*, vol. 13, no. 1, Mar. 2023, doi: 10.1038/s41598-023-30941-0.

[20] I. Saha, N. Ghosh, D. Maity, A. Seal, and D. Plewczynski, "COVID-DeepPredictor: Recurrent neural network to predict SARS-COV-2 and other pathogenic viruses," *Frontiers in Genetics*, vol. 12, Feb. 2021, doi: 10.3389/fgene.2021.569120.

[21] M. A. El-dosuky, M. Soliman, and A. E. Hassanien, "COVID-19 vs influenza viruses: A cockroach optimized deep neural network classification approach," *International Journal of Imaging Systems and Technology*, vol. 31, no. 2, pp. 472–482, Feb. 2021, doi: 10.1002/ima.22562.

[22] S. Kar and M. Ganguly, "Application of genomic signal processing as a tool for high-performance classification of SARS-CoV-2 variants: a machine learning-based approach," *Soft Computing*, vol. 28, no. 4, pp. 2891–2918, Jan. 2024, doi: 10.1007/s00500-023-09577-9.

[23] M. Togrul and H. Arslan, "Detection of SARS-CoV-2 Main Variants of Concerns using Deep Learning," *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, Sep. 2022, doi:10.1109/asyu56188.2022.9925559.

[24] S. Basu and R. H. Campbell, "Classifying COVID-19 variants based on genetic sequences using deep learning models," in *Springer series in reliability engineering*, 2022, pp. 347–360. doi: 10.1007/978-3-031-02063-6$_1$9.$\backslash NCBIVirus,''Available : https : //www.ncbi.nlm.nih.gov/labs/ virus/vssi/$.

[25] S. S. Sahu and G. Panda, "Identification of Protein- Coding regions in DNA sequences using a Time-Frequency filtering approach," Genomics, *Proteomics and Bioinformatics*, vol. 9, no. 1–2, pp. 45–55, Apr. 2011, doi: 10.1016/s1672-0229(11)60007-7.

[26] M. Akhtar, J. Epps and E. Ambikairajah, "Signal Processing in Sequence Analysis: Advances in Eukaryotic Gene Prediction," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 3, pp. 310-321, June 2008, doi: 10.1109/JSTSP.2008.923854.

[27] M. Akhtar, J. Epps and E. Ambikairajah, "On DNA Numeric Representations for Period-3 Based Exon Prediction," 2007 IEEE International Workshop on Genomic Signal Processing and Statistics, Tuusula, Finland, 2007, pp. 1-4, doi: 10.1109/GENSIPS.2007.4365821.

[28] N. Ahmed, T. Natarajan and K. R. Rao, ”Discrete Cosine Transform,” in *IEEE Transactions on Computers*, vol. C-23, no. 1, pp. 90-93, Jan. 1974, doi: 10.1109/T-C.1974.223784.

[29] M. Kumar and T. K. Rawat, “Design of fractional order differentiator using type-III and type-IV discrete cosine transform,” Engineering Science and Technology, an International Journal, vol. 20, no. 1, pp. 51–58, Feb. 2017, doi: 10.1016/j.jestch.2016.07.002

[30] M. Hassanzadeh and B. Shahrrava, ”Linear Version of Parseval’s Theorem,” in *IEEE Access*, vol. 10, pp. 27230-27241, 2022, doi: 10.1109/ACCESS.2022.3157736.

[31] Patrick J. Van Fleet, ”THE HAAR WAVELET TRANSFORMATION,” in Discrete Wavelet Transformations: An Elementary Approach with Applications , *Wiley*, 2019, pp.125-181, doi: 10.1002/9781119555414.ch4.

[32] Shyh-Jier Huang and Cheng-Tao Hsieh, ”Coiflet wavelet transform applied to inspect power system disturbance-generated signals,” in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 204-210, Jan. 2002, doi: 10.1109/7.993240.

[33] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philosophical Transactions - Royal Society. Mathematical, Physical and Engineering Sciences/Philosophical Transactions - Royal Society. Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, Apr. 2016, doi: 10.1098/rsta.2015.0202.

[34] E. Elhaik, “Principal Component Analyses (PCA)-based findings in population genetic studies are highly biased and must be reevaluated,” *Scientific Reports*, vol. 12, no. 1, Aug. 2022, doi: 10.1038/s41598-022-14395-4.

[35] N. Zhao, W. Mio and X. Liu, ”A hybrid PCA-LDA model for dimension reduction,” The 2011 International Joint Conference on Neural Networks, San Jose, CA, USA, 2011, pp. 2184-2190, doi: 10.1109/IJCNN.2011.6033499.

[36] E. I. G. Nassara, E. Grall-Maës and M. Kharouf, ”Linear Discriminant Analysis for Large-Scale Data: Application on Text and Image Data,” 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 2016, pp. 961-964, doi: 10.1109/ICMLA.2016.0173.

[37] S. Kumari, D. Kumar, and M. Mittal, “An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier,” *International Journal of Cognitive Computing in Engineering*, vol. 2, pp. 40–46, Jun. 2021, doi: 10.1016/j.ijcce.2021.01.001.

[38] “ML-LGBM: A Machine Learning Model Based on Light Gradient Boosting Machine for the Detection of Version Number Attacks in RPL-Based Networks IEEE Journals and Magazine IEEE Xplore,” ieeexplore.ieee.org. doi: https://ieeexplore.ieee.org/document/9448047/references.

[39] V. Jackins, S. Vimal, M. Kaliappan, and M. Y. Lee, “AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes,” *The Journal of Supercomputing*, Nov. 2020, doi: https://doi.org/10.1007/s11227-020-03481-x.

[40] J. Hatwell, M. M. Gaber, and R. M. Atif Azad, “Ada-WHIPS: explaining AdaBoost classification with applications in the health sciences,” *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, Oct. 2020, doi: https://doi.org/10.1186/s12911-020-01201-2.

[41] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986, doi: https://doi.org/10.1007/bf00116251.

[42] S. M. I. Osman and A. Sabit, ”Predictors of COVID-19 Vaccination Rate in USA: A Machine Learning Approach,” *Machine Learning with Applications*, vol. 10, Dec. 2022, Art. no. 100408. [Online]. Available: https://doi.org/10.1016/j.mlwa.2022.100408

[43] M.A.Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, ”Support vector machines,” in *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18-28, July-Aug. 1998, doi: 10.1109/5254.708428.

[44] T. M. T. A. Hamid, R. Sallehuddin, Z. M. Yunos, and A. Ali, ”Ensemble Based Filter Feature Selection with Harmonize Particle Swarm Optimization and Support Vector Machine for Optimal Cancer Classification,” *Machine Learning with Applications*, vol. 4, Mar. 2021, Art. no. 100054. Available: https://doi.org/10.1016/j.mlwa.2021.100054