

An Efficient Method For Classification of different types Of Hepatitis Virus using Genomic Signal Processing and Machine Learning

1st Love Fadia

*Electrical and Computer Engineering
University of Windsor
Windsor, Canada
fadial@uwindsor.ca*

2nd Vatsal Shah

*Electrical and Computer Engineering
University of Windsor
Windsor, Canada
shah7r1@uwindsor.ca*

3rd Mohammad Hassanzadeh*

*Electrical and Computer Engineering
University of Windsor
Windsor, Canada
mhassan@uwindsor.ca*

*Corresponding author

4th , Jonathan Wu

*Electrical and Computer Engineering
University of Windsor
Windsor, Canada
jwu@uwindsor.ca*

5th Majid Ahmadi

*Electrical and Computer Engineering
University of Windsor
Windsor, Canada
M.ahmadi@uwindsor.ca*

Abstract—Hepatitis virus is one of the deadliest and it causes loss of several individuals life. This paper describes an efficient method for classifying the four main hepatitis virus types—Hepatitis-B, Hepatitis-C, Hepatitis-D, and Hepatitis-E using Genomic Signal Processing and Machine Learning. First, we gather the dioxy ribonucleic acid sequences for the various strains of the Hepatitis virus. Next, we convert this sequences from characters to numbers using a variety of coding algorithms. Subsequently, We employ some well-known signal processing methodologies along with some modified versions of signal processing developed by ourselves to extract the characteristics from the transformed sequences and then we use Singular Value Decomposition for dimensionality reduction. Ultimately, two machine learning models—Decision Tree and Light Gradient Boosting Machine are trained for classification. Our approach gets an accuracy of 99% with the combination of normalisation of atomic number with a customised Haar Wavelet.

Index Terms—Genomic signal processing, Fast Fourier Transform, Haar Wavelet Transform, Coiflett Wavelet Transform, Machine Learning algorithms.

I. INTRODUCTION

Classifying variants of the Hepatitis virus is essential as it causes liver cancer and due to that millions of people die every year. Researchers are developing a system that takes the blood sample, and then based on dioxy ribonucleic acid Sequences (DNA) they do analysis and develop the vaccine. However, this method is costly and requires many expensive devices [1]. Therefore, it is essential to have a non-invasive approach towards it by taking data from an open-source biological database and performing analysis using signal processing techniques to generate features. This feature helps to distinguish among various strains of this virus as similar as identifying different mutation patterns in the DNA sequences of the various subtypes of the virus [2]. With the

help of machine learning, the task of classification becomes easier. Deep learning-based unsupervised clustering of DNA sequences has been proposed by article [3]. In this authors performed the Frequency chaos game representation technique (FCGR) and plotted this matrix as an image. After this, a deep learning model was employed and the classification of Hepatitis virus, Influenza virus, and Dengue was accomplished efficiently. An efficient hybrid predictive model has also been developed [4] which enables the experts to classify Hepatitis virus C (HCV). The deep learning also helps medical experts to identify different genotypes of HCV [5] Although it was 10 class classification the limitation is class imbalance here each types of genotype had not same amount of DNA sequence. Clinical data-based classification of Hepatitis B (HBV) virus from normal one was proposed in [6]. Here certain tests were performed on patients and based on those parameters model is trained and after that classification of the HBV virus takes place. Using 12 features of the clinical dataset with feature selection and without it, the classification of HCV virus had been proposed [7] using machine learning classifiers like Naïve Byes (NB), K-nearest neighbor (KNN), Random forest (RF), and logistic regression (LR). Based on fuzzy networks classification of Hepatitis virus is also possible [8] But the limitation is there is limited amount of data in that.

The features of the article are: A novel approach to assigning numbers to DNA has been proposed in which using atomic number normalization each nucleotide is assigned specific numbers. A little modification in Haar wavelet transform is implemented which also outperforms traditional Haar wavelet in two numbering schemes.

The table 1 shows the comparison with other state of art methods.

TABLE I
COMPARISON WITH STATE OF THE ART METHODS

Sr No	Author Name	Methodology	Dataset	Accuracy (%)
1	Arias et al.	Unsupervised Clustering of DNA Sequence + Deep Learning	Influenza (949), Dengue (1633), Hepatitis B (1562)	99.6
2	Lilhore et al.	Hybrid model for classification using improved SVM and Random Forest	UCI Database consisting clinical data of Hepatitis C	96.67
3	M Fahmy et al.	FCGR + Deep Learning	10 genotypes of Hepatitis C	99
4	Nilashi et al.	Fuzzy Neural Network	155 records: 32 die (20.6%) and 123 live (79.4%) cases	100
5	Proposed Methodology	3 Coding Techniques + 3 Signal Processing Techniques: Haar Wavelet, Customised Haar Wavelet, FFT	Hepatitis B (998), Hepatitis C (998), Hepatitis D (998), Hepatitis E (998)	99

From the state of art comparison it is clear that article 2 has class imbalance while in 3 and 4 authors are performing one class classification but in our approach we have not imbalance issue and we did classification of 4 different types of Hepatitis virus.

II. METHODOLOGY

We have collected data on Hepatitis B, C, D, and E from the National Centre of Biotechnology Institute (NCBI) website which is an open source tool where different organisation all over the world publishes the DNA sequence that has been generated. It consists of the sequences ranging from Dengue to COVID-19. We gathered 1000 sequences of each subtype of the Hepatitis virus having a length of 1000. After that, it has been converted into three numeric schemes and once we get different numeric sequences we apply signal processing techniques that help generate features. Then we applied Singular value Decomposition (SVD) which helps to reduce the dimension of the long features while retaining as much information as it can and at last machine learning classifiers were implemented for classification. The flowchart illustrates the same.

A. Coding Technique

Here we implemented three coding techniques which basically converts DNA sequences into numbers.

1) *EIIP coding*: The DNA sequence can be converted into floating points using the Electron-Ion potential value assigned to the nucleotide of the sequence which are [9]:

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

2) *Normalisation of atomic number*: Another approach that we implemented is changing the DNA sequence into floating point based on the atomic number of each nucleotide present in the sequence once we assigned numbers then we normalized it. The number that each nucleotide assigned is:

- Adenine (A): 0.257 • Guanine (G): 0.287
- Thymine (T): 0.243 • Cytosine (C): 0.213

3) *Complex coding*: Based on the complementary property of DNA sequence one can also apply complex conjugate numbers to the nucleotide [10]:

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

B. Signal Processing

1) *Discrete Sine Transform*: The Discrete Sine Transform (DST) represents the sinusoidal terms of the Fourier transform of varying frequency. It has good energy compaction and it has 4 different versions but we utilized version II which is explained in the equation [11]:

$$X_k = 2 \sum_{n=0}^{N-1} x_n \sin \left(\frac{\pi(n+0.5)(k+1)}{N} \right), \text{ for } k = 0, 1, \dots, N-1 \quad (1)$$

- X_k : The k -th coefficient of the Discrete Sine Transform, representing the amplitude of the sine wave at frequency k .
- x_n : The n -th element of the input sequence, representing the original data point or signal value at position n .
- n : An index variable ranging from 0 to $N-1$, representing the position in the input sequence.
- k : An index variable ranging from 0 to $N-1$, representing the frequency index in the transformed sequence.
- N : The total number of points in the input sequence, representing the length of the sequence being transformed.
- $\frac{\pi}{N}$: A scaling factor in the argument of the sine function, representing the angular frequency step.

2) *Haar Wavelet Transform*: Haar wavelet transform is the function of rescaled square shapes. It is one of the simplest wavelet transforms. Here we implemented this transformation at level 4. The equation can be also represented in the form of a matrix that is [12]:

$$H_4 = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

Where first row represents scaling coefficients and from second to third level it represents various levels coefficients.

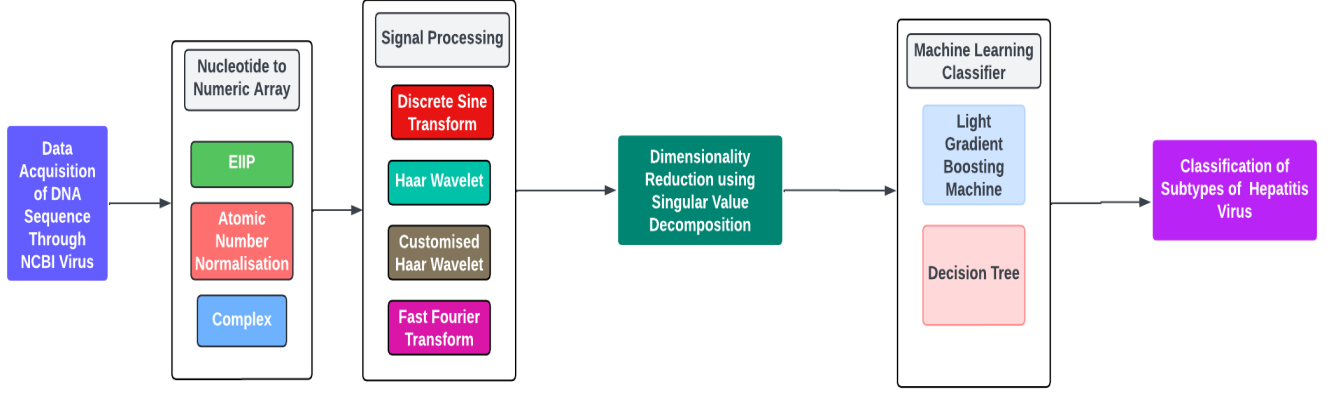


Fig. 1. Methodology for Classification of Hepatitis Virus using Signal Processing and Machine Learning

3) *Customised Haar wavelet Transform*: Here we customised the coefficients of haar wavelet transform and we get better accuracy the customisation is represented in form of matrix as:

Given a signal vector $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}$, the custom wavelet transform matrix at Level 1 can be represented as:

$$W = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

It can be realised using structure figure 2:

This matrix can be applied to the signal \mathbf{x} to produce the wavelet coefficients.

4) *Fast Fourier Transform*: The Fast Fourier Transform (FFT) is the algorithm that computes discrete fourier transform on the signal at optimal rate. The equation can be stated as [13]:

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

C. Dimensionality Reduction

Singular Value Decomposition (SVD) is a mathematical technique used to factorize a matrix into three distinct matrices: U , Σ , and V^T . Given a matrix A , SVD expresses it as

$$A = U\Sigma V^T \quad (3)$$

where U and V are orthogonal matrices representing the left and right singular vectors, respectively, and Σ is a diagonal matrix containing the singular values of A . These singular values represent the magnitude of the directions defined by the singular vectors [14]. SVD is widely used in data compression, noise reduction, and dimensionality reduction, as it allows for

Step 1: Multiply with Haar wavelet coefficients
Step 2: Decomposing ahead to Level 3 and 4

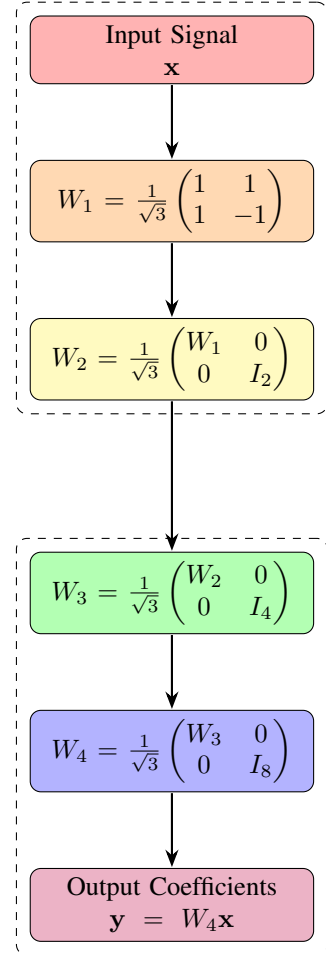


Fig. 2. Customised Haar Wavelet Transform at Level 4

the approximation of a matrix by retaining only the largest singular values and their corresponding vectors, effectively capturing the most significant features of the data. Here we have taken the U matrix as input for the classifier model we have taken the first 5 rows of the U matrix. The reason for choosing the U matrix is because it has eigen vectors rather than values and it is found that more significant variations are stored in the U matrix [15].

D. Machine Learning

1) *Light Gradient Boosting Machine*: Using the idea of boosting, an ensemble learning technique called a Light Gradient Boosting Machine (LGBM) is employed. The LGBM model operates on a similar principle to other boosting algorithms. The model successively builds an ensemble of decision trees while ensuring that each tree gains knowledge from the errors of the trees built in earlier iterations. In the end, the weighted total of all the decision trees that were built is used to create the final model. Moreover, unlike traditional level-wise tree development, which increases the tree's levels one at a time, LGBM uses a leaf-wise growth strategy. Stated differently, it causes the tree to grow by increasing the size of the leaf that reduces loss the most. This method may produce deeper trees with fewer nodes, increasing efficiency and accuracy [16]. Besides this, an additional technique that LightGBM uses to increase speed and efficiency is a histogram-based algorithm that divides continuous feature values into discrete bins. This makes it easier to identify the best-split sites inside the trees, which greatly accelerates training, especially for huge datasets [17].

2) *Decision Tree*: A popular machine-learning approach for many classification and regression tasks is the decision tree. A decision tree breaks down a complex problem into smaller alternatives that build up to a conclusion or forecast in an attempt to mimic how people actually make decisions. In the tree structure, a node represents a feature-based choice, a branch represents the decision's outcome, and a leaf represents the final product. The tree begins in step one with a root node that symbolizes the whole dataset. Next, each feature is evaluated by the algorithm to determine which best partitions the data into distinct groups for classification [18]. Based on a number of selection criteria, including Gini Impurity and Information Gain, the optimal feature to split the node is chosen. In addition, the dataset is subdivided into branches from the root node when the best feature is selected using the previously selected selection criteria. Each branch shows how the feature might turn out. fresh nodes are created following each branch to reflect decisions made in the context of fresh knowledge. The process of separating data points is carried out recursively until the algorithm either perfectly classifies all data points or meets a predefined stopping threshold. The leaf node that appears after each branch represents the ultimate output, which is a class label in classification [19].

III. EXPERIMENTAL RESULTS

The accuracy of two classifiers, LGBM and Decision Tree, is compared in Table I for a range of signal processing and coding strategies. In every test situation, LGBM consistently outperforms the Decision Tree. Impressively, the accuracies for LGBM often reach 99%, especially when using the Discrete Sine Transform and Custom Haar Wavelet under Complex and EIIP coding schemes. Although useful, the Decision Tree's accuracy is lower, especially when dealing with complex code, where it only reaches 97% accuracy. Notably, EIIP and normalization of atomic number (NAN) coding considerably improve signal characteristics for classification; both classifiers perform best with these techniques. With EIIP coding, both classifiers perform well, scoring perfect or almost perfect accuracy in cases requiring the Fast Fourier Transform and Discrete Sine Transform. This pattern indicates that EIIP coding, particularly when combined with particular signal processing methods like wavelets and Fourier transformations, successfully highlights important features for these classifiers. The LGBM classifier is durable and versatile for a variety of data transformations, as evidenced by its outstanding consistency, which keeps it at high accuracy across a wide range of coding and signal processing combinations. While a little less reliable, the Decision Tree's performance is still rather good, especially when it comes to normalizing atomic number NAN coding, where it performs exceptionally well when using the Discrete Sine Transform and Haar Wavelet Transform. Although it does so marginally less well than LGBM, this robustness highlights the classifier's ability to handle complicated feature sets. The information as a whole suggests that selecting the appropriate mix of coding and signal processing methods can have a big impact on classification accuracy. While the Decision Tree provides robustness with certain configurations, especially when combined with EIIP and coding, LGBM consistently performs well over a wide range of techniques, making it a better option for complicated datasets. When choosing coding and signal processing techniques for applications that call for complex data analysis and categorization, these insights may be helpful.

A. Comparative Analysis of Coding Techniques

Using combined data from LGBM and Decision Tree classifiers, Table II offers a consolidated examination of classification accuracies across three coding techniques: complicated, EIIP, and NAN. The complicated technique has the lowest average accuracy, at 95.38%, and the highest standard deviation, at 0.030, suggesting a substantial degree of variability in classifier performance. On the other hand, the EIIP coding method yields better average accuracy of 96.88% and a little smaller standard deviation of 0.026, indicating more consistent outcomes in various configurations. Furthermore, with the lowest standard deviation of 0.017 and an average accuracy of 96.75%, the NAN approach demonstrates comparable performance and yields the most consistent results among the three. While all strategies have the lowest recorded accuracy not going below 92%, NAN normalization has a slightly superior

TABLE II
CLASSIFIER ACCURACIES BY SIGNAL PROCESSING TECHNIQUE AND CODING METHOD

Sr No	Coding Technique	Signal Processing	Classifier	Accuracy (%)
1	Complex	DST	LGBM	94%
2	Complex	Custom Haar Wavelet	LGBM	99%
3	Complex	Haar Wavelet Transform	LGBM	99%
4	Complex	FFT	LGBM	98%
5	EIIP	DST	LGBM	99%
6	EIIP	FFT	LGBM	99%
7	EIIP	Haar Wavelet Transform	LGBM	97%
8	EIIP	Custom Haar Wavelet	LGBM	98%
9	NAN	Haar Wavelet Transform	LGBM	98%
10	NAN	DST	LGBM	98%
11	NAN	FFT	LGBM	98%
12	NAN	Custom Haar Wavelet	LGBM	99%
13	Complex	DST	Decision Tree	92%
14	Complex	Custom Haar Wavelet	Decision Tree	92%
15	Complex	Haar Wavelet Transform	Decision Tree	92%
16	Complex	FFT	Decision Tree	97%
17	EIIP	DST	Decision Tree	99%
18	EIIP	FFT	Decision Tree	98%
19	EIIP	Haar Wavelet Transform	Decision Tree	93%
20	EIIP	Custom Haar Wavelet	Decision Tree	92%
21	NAN	Haar Wavelet Transform	Decision Tree	94%
22	NAN	DST	Decision Tree	97%
23	NAN	FFT	Decision Tree	95%
24	NAN	Custom Haar Wavelet	Decision Tree	95%

accuracy of 94%. The maximum recorded accuracy is shared by both the EIIP and NAN coding procedures, at 99%. These numbers indicate that although high accuracy may be achieved by all three coding schemes, somewhat better performance and greater reliability are offered by EIIP and NAN when compared to the complicated methodology. This investigation highlights how machine learning models in a variety of signal processing scenarios can benefit from improved predicted accuracy and stability through the use of EIIP and NAN coding approaches.

TABLE III
CONSOLIDATED PERFORMANCE ANALYSIS OF CODING TECHNIQUES

Coding Technique	Avg. Acc. (%)	Std. Dev	Worst Acc. (%)	Best Acc. (%)
Complex	95.38	0.030	92	99
EIIP	96.88	0.026	92	99
NAN	96.75	0.017	94	99

B. Comparative Analysis of Signal Processing Techniques

Table III offers a thorough examination of how well four signal processing methods perform when applied to various coding strategies and classifiers: DST, Custom Haar Wavelet, Haar Wavelet Transform, and Fast Fourier Transform. For each signal processing method, the average accuracy, standard deviation, worst accuracy, and greatest accuracy are summarized in each row. With a comparatively low standard deviation of 0.027 and the highest average accuracy of 95.50%, the FFT exhibits consistent performance across many contexts. With a

standard deviation of 0.033 and an average accuracy of 95.38% and 95.25%, respectively, the Custom Haar Wavelet and Haar Wavelet Transform approaches both indicate a little bit more fluctuation in their performance. Analogously, the DST performs consistently, with an average accuracy of 95.25% and a standard deviation of 0.027, comparable to the FFT. With a range of accuracy from 92% in the worst case to 99% in the best case, all strategies indicate their potential efficacy in certain scenarios. According to the results, all methods work well, but FFT stands out as having somewhat better average accuracy and consistency, which makes it a great option for tasks requiring good performance in signal processing.

TABLE IV
CONSOLIDATED PERFORMANCE ANALYSIS BY SIGNAL PROCESSING TECHNIQUE

Signal Processing	Avg. Acc. (%)	Std. Dev	Worst Acc. (%)	Best Acc. (%)
Discrete Sine Transform	95.25	0.027	92	99
Custom Haar Wavelet	95.38	0.033	92	99
Haar Wavelet Transform	95.25	0.033	92	99
Fast Fourier Transform	95.50	0.027	92	99

C. Comparative Analysis of Machine Learning Classifier

A thorough comparison of the two classifiers' performances—LGBM and Decision Tree—is given in Table IV. With an average accuracy of 98.00%, LGBM performs well, proving its dependability in a variety of situations. With a standard deviation of only 0.014, LGBM is comparatively accurate, indicating that it doesn't change in accuracy according to the coding or signal processing methods employed. LGBM can maintain high accuracy under a variety of scenarios; its peak performance is 99%, while its worst-case accuracy is 94%. Conversely, the Decision Tree classifier displays an average accuracy of 94.67%, which is marginally lower. Its performance appears to be more variable than LGBM, as seen by its greater standard deviation of 0.025. Though its worst-case accuracy falls to 92%, indicating a wider range in performance, the Decision Tree nevertheless attains a 99% best-case accuracy, which is comparable to LGBM. For the datasets and methods under analysis, LGBM is the most reliable classifier since it performs better overall than Decision Tree in terms of average accuracy and consistency. Even if it is a little less reliable, the Decision Tree is still a competitive choice, especially when it attains nearly optimal accuracy.

TABLE V
CONSOLIDATED PERFORMANCE ANALYSIS BY CLASSIFIER

Machine Classifier	Avg. Acc. (%)	Std. Dev	Worst Acc. (%)	Best Acc. (%)
LGBM	98.00	0.014	94	99
Decision Tree	94.67	0.025	92	99

IV. CONCLUSION

This study evaluated the effectiveness of signal processing and coding methodologies with LGBM and Decision Tree classifiers. The results demonstrate the dependability of LGBM in many settings, as it continuously outperformed the Decision Tree with an average accuracy of 98.00% and negligible variation. With an average accuracy of 94.67 the Decision Tree showed more variability despite its usefulness. FFT was shown to be the most efficient signal processing technique, with an average accuracy of 95.50. Coding strategies like EIIP and NAN normalization, with best-case accuracies of 99%, yielded the best results when paired with FFT and LGBM. The combination of LGBM with FFT and EIIP/NAN normalization coding has to be the most accurate and consistent approach for classification jobs in signal processing.

REFERENCES

- [1] T. Tu, K. Patel, and N. A. Shackel, "Viral hepatitis," in *Elsevier eBooks*, pp. 317–340, 2017. doi: 10.1016/b978-0-12-800685-6.00017-5.
- [2] E. Dougherty, X. Cai, Y. Huang, S. Kim, and R. Yamaguchi, "Genomic Signal Processing," *Current Genomics*, vol. 10, no. 6, p. 364, Sep. 2009, doi: 10.2174/138920209789177593.
- [3] P. M. Arias, F. Alipour, K. A. Hill, and L. Kari, "DeLUCS: Deep learning for unsupervised clustering of DNA sequences," *PLoS ONE*, vol. 17, no. 1, p. e0261531, Jan. 2022, doi: 10.1371/journal.pone.0261531.
- [4] U. K. Lilhore et al., "Hybrid model for precise hepatitis-C classification using improved random forest and SVM method," *Scientific Reports*, vol. 13, no. 1, Aug. 2023, doi: 10.1038/s41598-023-36605-3.
- [5] A. M. Fahmy, M. S. Hammad, M. S. Mabrouk, and W. I. Al-Atabany, "On leveraging self-supervised learning for accurate HCV genotyping," *Scientific Reports*, vol. 14, no. 1, Jul. 2024, doi: 10.1038/s41598-024-64209-y.
- [6] A. J. Mueller-Breckenridge et al., "Machine-learning based patient classification using Hepatitis B virus full-length genome quasiespecies from Asian and European cohorts," *Scientific Reports*, vol. 9, no. 1, Dec. 2019, doi: 10.1038/s41598-019-55445-8.
- [7] H. Mamdouh, M. Y. Shams, and T. A. El-Hafeez, "Hepatitis C Virus Prediction Based on Machine Learning Framework: a Real-world Case Study in Egypt," *Research Square (Research Square)*, Jan. 2022, doi: 10.21203/rs.3.rs-1292024/v1.
- [8] M. Nilashi, H. Ahmadi, L. Shahmoradi, O. Ibrahim, and E. Akbari, "A predictive method for hepatitis disease diagnosis using ensembles of neuro-fuzzy technique," *Journal of Infection and Public Health*, vol. 12, no. 1, pp. 13–20, Jan. 2019, doi: 10.1016/j.jiph.2018.09.009.
- [9] 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.
- [10] 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.
- [11] V. Britanak, P. C. Yip, and K. R. Rao, "Fast DCT/DST algorithms," in *Elsevier eBooks*, 2007, pp. 73–140. doi: 10.1016/b978-012373624-6/50006-0.
- [12] 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.
- [13] M. Hassanzadeh and B. Shahrava, "Linear Version of Parseval's Theorem," in *IEEE Access*, vol. 10, pp. 27230–27241, 2022, doi: 10.1109/ACCESS.2022.3157736.
- [14] V. Klema and A. Laub, "The singular value decomposition: Its computation and some applications," in *IEEE Transactions on Automatic Control*, vol. 25, no. 2, pp. 164–176, April 1980, doi: 10.1109/TAC.1980.1102314.
- [15] D. Schmidt, "A Survey of Singular Value Decomposition Methods for Distributed Tall/Skinny Data," *2020 IEEE/ACM 11th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)*, GA, USA, 2020, pp. 27–34, doi: 10.1109/ScalA51936.2020.00009.
- [16] "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](https://ieeexplore.ieee.org/document/9448047/references).
- [17] N. Aziz, E. A. P. Akhir, I. A. Aziz, J. Jaafar, M. H. Hasan and A. N. C. Abas, "A Study on Gradient Boosting Algorithms for Development of AI Monitoring and Prediction Systems," *2020 International Conference on Computational Intelligence (ICCI)*, Bandar Seri Iskandar, Malaysia, 2020, pp. 11–16, doi: 10.1109/ICCI51257.2020.9247843.
- [18] 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>.
- [19] Y.-Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," *PubMed Central (PMC)*, Apr. 2015, doi: 10.11919/j.issn.1002-0829.215044.