viruses

MDPI

Article

The Utility of Data Transformation for Alignment, De Novo Assembly and Classification of Short Read Virus Sequences

Avraam Tapinos [1,*], Bede Constantinides [1,2], My V. T. Phan [3], Samaneh Kouchaki [1,4], Matthew Cotten [3,5,6] and David L. Robertson [1,5]

[1] School of Biological Sciences, The University of Manchester, Manchester M13 9PT, UK; bede.constantinides@manchester.ac.uk (B.C.); samaneh.kouchaki@eng.ox.ac.uk (S.K.); david.l.robertson@glasgow.ac.uk (D.L.R.)
[2] Modernising Medical Microbiology Consortium, Nuffield Department of Clinical Medicine, John Radcliffe Hospital, University of Oxford, Oxford OX3 9DU, UK
[3] Department of Viroscience, Erasmus Medical Centre, Doctor Molewaterplein 40, 3015 GD Rotterdam, The Netherlands; v.t.m.phan@erasmusmc.nl (M.V.T.P.); mlcotten13@gmail.com (M.C.)
[4] Department of Engineering Science, Institute of Biomedical Engineering, University of Oxford, Oxford OX3 7DQ, UK
[5] MRC-University of Glasgow Centre for Virus Research, Glasgow G61 1QH, UK
[6] MRC/UVRI & LSHTM Uganda Research Unit Entebbe, P.O. Box 49 Entebbe, Uganda
[*] Correspondence: avraam.tapinos@manchester.ac.uk; Tel.: +44-(0)-161-701-7563

Received: 30 March 2019; Accepted: 22 April 2019; Published: 26 April 2019

Abstract: Advances in DNA sequencing technology are facilitating genomic analyses of unprecedented scope and scale, widening the gap between our abilities to generate and fully exploit biological sequence data. Comparable analytical challenges are encountered in other data-intensive fields involving sequential data, such as signal processing, in which dimensionality reduction (i.e., compression) methods are routinely used to lessen the computational burden of analyses. In this work, we explored the application of dimensionality reduction methods to numerically represent high-throughput sequence data for three important biological applications of virus sequence data: reference-based mapping, short sequence classification and de novo assembly. Leveraging highly compressed sequence transformations to accelerate sequence comparison, our approach yielded comparable accuracy to existing approaches, further demonstrating its suitability for sequences originating from diverse virus populations. We assessed the application of our methodology using both synthetic and real viral pathogen sequences. Our results show that the use of highly compressed sequence approximations can provide accurate results, with analytical performance retained and even enhanced through appropriate dimensionality reduction of sequence data.

Keywords: alignment; assembly; taxonomic classification; time series; data transformation; DWT; DFT; PAA; data compression; compressive genomics

1. Introduction

Next-generation sequencing (NGS) enables massively parallel determination of nucleotide order within genetic material, making it possible to rapidly sequence the genomes of individuals, populations and metagenomic samples [1–5]. However, the sequences generated by these instruments are almost always considerably shorter in length than the genomic regions studied. Genomic analyses often begin with the process of sequence assembly, where sequence fragments (reads) are reconstructed into the larger sequences from which they originated. Computational methods play a vital role in the assembly of short reads, and a variety of assemblers and related tools have been developed in tandem with emerging sequencing platforms [6]. All subsequent analyses and investigations depend upon the quality, accuracy and speed of this crucial sequence assembly process.

There are many computational methods to generate consensus sequences representing the genomes of species in a sample. Such approaches include seed-and-extend alignment methods using suffix array derivatives, such as the Burrows-Wheeler Transform (BWT) for aligning short reads informed by a known reference sequence [7,8], graph-based methods employing Overlap Layout Consensus (OLC) [9,10] and de Bruijn graphs of *k*-mers [11–13] for reference-free de novo sequence assembly. However, for sequencing projects to characterise genetic variation within populations (deep sequencing), metagenomics and pathogen discovery, the effectiveness of the aforementioned approaches varies considerably [14].

Samples with mixed viral infections, especially those comprising divergent variants, present a number of analytical and computational problems. The use of a reference sequence, even the use of a data specific generated sequence, can lead to valuable read information being discarded during the alignment process [15]. On the other hand, while de novo approaches require little a priori knowledge of target sequence composition, the methods are computationally intensive, and their performance scales poorly with datasets of increasing size [9]. Aggressive heuristics must be employed, to traverse graphs and deal with mismatches, reduce the running time of de novo assemblers, which, in turn, can compromise assembly quality. Indexing structures such as the BWT and its relatives are widely used to reduce the burden of pairwise sequence comparison, for both reference-based mapping and de novo assembly. However, they cannot process mismatches within reads, necessitating the use of computationally expensive heuristics to establish relationships between divergent sequences. Increasing sequence length further affects the performance of these approaches [16].

A major challenge in working with NGS data from metagenomic studies is the high levels of diversity present, particularly for the virus genetic material. Also, the number of sequences generated challenge many computational systems for a feasible working solution in terms of time and the computational resources typically available in biological laboratories. For biologists working on outbreak responses or pathogen discovery, both the accuracy of the assembly results and the speed of sequence analyses (e.g., assembly, alignment and pathogen classification) are crucial for crisis response and management. The ability to run analyses in the field on portable computer systems without internet connectivity is also important. Here, we explore the utility of data transform methods to extract major features from viral NGS sequence data and use the features to analyse data in a lower dimensional space.

Similar analytical challenges involving high dimensional sequential data are encountered in other data-intensive fields, such as signal and image processing, and time series analysis, where data transforms and approximation techniques are used for data dimensionality reduction. Data transform/approximation techniques include the discrete Fourier transform (DFT) [17], the discrete wavelet transform (DWT) [18,19] and piece-wise aggregate approximation (PAA) [20,21]. The DFT or DWT are used to transform data to their frequency domains, allowing feature extraction [22], and PAA is used as a data approximation approach. In data-intensive fields, data transformations/approximations are commonly used as dimensionality reduction approaches for obtaining fast approximate solutions for a given problem. Due to the ordered nature of genetic data, many of these transformation approaches can be applied to sequences of nucleotides [23] or amino acids [24]. An example of a successful implementation of a Fourier transform in computational biology is the multiple sequence alignment based on fast Fourier transform alignment algorithm MAFFT [25] where the physiochemical properties of amino acids are used to represent sequences for fast matching of homologous sequence regions for alignment. Since most transformation approaches are suitable only for numerical sequences, the strings of letters representing genetic sequences must be mapped into numerical space using a numerical sequence representation method [26].

In addition to the DFT, the DWT and PAA, suitable methods for measuring the pairwise similarity of sequential data or transformations include the Lp-norms [27], dynamic time warping (DTW) [28], longest common subsequence (LCS) [29], and alignment approaches, such as the Needleman-Wunsch and Smith-Waterman algorithms. Euclidean distance is arguably the most widely used Lp-norm method for sequential data comparison but can only be used on sequences of the same length. Furthermore, Lp-norm methods do not accommodate shifts in the *x*-axis (time or position) and are thus limited in their ability to identify similar features within offset data. Elastic similarity/dissimilarity methods, such as LCS, unbounded DTW and various alignment algorithms, permit comparison of data with different dimensions and tolerate shifts in the *x*-axis. These properties of elastic similarity methods can be very useful in the analysis of speech signals, for example, but can be computationally expensive [30]. Several approaches have been proposed to permit fast searching with DTW, including the introduction of global constraints (wrapping path) or the use of lower bounding techniques, such as LB_keogh [28].

While pairwise comparison methods may be used for clustering, classification and similarity searches, they are very time consuming for large datasets ($O(n^2)$ time complexity). Indexing structures, such as the *R**-tree, *KD*-tree, *VP*-tree and *MVP*-tree have significantly lower time complexity ($O(n \log(n))$) for similarity search [31] and are more appropriate for efficient analysis of large datasets. The *R**-tree [32,33] and *KD*-tree [34] indexing structures are very accurate for low dimensional datasets. However, their performance deteriorates significantly in high dimensional space [31], a phenomenon known as the 'curse of dimensionality' [35,36]. Metric trees, such as the *VP*-tree [37] and *MVP*-tree [38], are less prone to this limitation. Metric space indexing structures make use of geometric properties for partitioning data and work efficiently on both low and high dimensional data [39]. The curse of dimensionality can be further mitigated using data approximations, such as the DFT, the DWT and the PAA, to partition a dataset in an approximated space without loss of generality [21].

Here, we investigate the performance of three established dimensionality reduction techniques on three common analysis tasks involving viral short read sequence data: classification, reference-based mapping/alignment and de novo assembly. We benchmarked the accuracy of our proposed methodology against existing tools, and demonstrate the applicability of time series and signal processing data mining techniques for the analysis of viral NGS data.

2. Materials and Methods

2.1. Symbolic to Numeric Sequence Representations

Various numeric sequence representation methods can be used for symbolising a nucleotide sequence to a numerical space (see 51). Depending on the chosen numerical representation, each nucleotide is associated with a specific numerical value or vector. The specific values are assigned to the position of each nucleotide indicating the presence of a nucleotide at each sequence position (Equation 1). R_i is the indicator for a specific nucleotide in the i^{th} position of the sequence S with a length of n nucleotides. Values $v_1 \ldots v_5$ correspond to the numerical value or numerical vector associated with each nucleotide.

$$R = \begin{cases} v_1 \; if \; i = A \\ v_2 \; if \; i = T \\ v_3 \; if \; i = C \\ v_4 \; if \; i = G \\ v_5 \; otherwise \end{cases} \quad \forall i \in S \tag{1}$$

Methods, such as the electron-ion interaction pseudopotentials (EIIP) [40] and the atomic representation approach [41], aim to mimic the biochemical properties of nucleic acids but introduce some mathematical bias that does not exist in reality [26]. Other methods, like the Voss indicator [42] and the Tetrahedron approach, do not introduce internucleotide mathematical bias, meaning the pairwise distances between each non-identical transformed nucleotide are the same (for example, the distance between A and T is equal to the distances between A and C as well as A and G). Furthermore, the cumulative sum of a numerical representation R can be used to indicate the trajectory of a sequence in nucleotide space. Table 1 indicates the values used for different representation methods [26].

Table 1. Numerical nucleotide sequence representation methods.

Method	Numerical Representation
Integer number	$A = 1,\ C = -1,\ G = 2,\ T = -2,\ N = 0$
Real number	$A = -1.5,\ C = 0.5, G = -0.5,\ T = 1.5,\ N = 0.0$
EIIP	$A = 0.1260,\ C = 0.1340,\ G = 0.0806,\ T = 0.1335,\ N = 0$
Atomic	$A = 70,\ C = 58,\ G = 78,\ T = 66,\ N = 0$
Pair	$A \textit{ or } T = 1,\ C \textit{ or } G = -1,\ N = 0$
Complex number	$A = 1 + 1i,\ C = -1 + 1i,\ G = -1 - 1i,\ T = 1 - 1i,\ N = 0 + 0i$
DNA Walk	$A = [1, 0],\ C = [0, 1],\ G = [0, -1],\ T = [-1, 0],\ N = [0, 0]$
Tetrahedron	$A = [0, 0, 1],\ C = \left[-\frac{\sqrt{2}}{3}, -\frac{\sqrt{6}}{3}, \frac{1}{3}\right],$ $G = \left[-\frac{\sqrt{2}}{3}, -\frac{\sqrt{6}}{3}, -\frac{1}{3}\right],\ T = \left[2 \times \frac{\sqrt{2}}{3}, 0, -\frac{1}{3}\right],\ N = [0, 0, 0]$
Voss indicator	$A = [0, 0, 1, 0],\ C = [1, 0, 0, 0],\ G = [0, 1, 0, 0],\ T = [0, 0, 0, 1],\ N = [0, 0, 0, 0]$

2.2. Sequence Transformation

Effective methods for transforming/approximating sequential data should: (i) accurately transform/approximate data without loss of useful information, (ii) have low computational overheads, (iii) facilitate rapid comparison of data and (iv) provide lower bounding—where the distance between data representations is always less than or equal to that of the original data—precluding false negative results [43]. The lower bounding property guarantees that if two data points are nearby in their original space, they will remain so in their transformed/approximate space. We employ the DFT and the DWT transformation methods and the PAA approximation method as they satisfy the above requirements, and these are widely used for analysing discrete signals [44] and can be used to transform/approximate nucleotide sequence numerical representations to different levels of resolution, permitting reduced dimensionality sequence analysis.

Figure 1A illustrates an example of the DFT and DWT transformations and PAA approximation of a short nucleotide sequence. The DFT and the fast Fourier transform (FFT) convert data from their original domain into the frequency domain. In principle, the DFT decomposes a numerically represented nucleotide sequence with n positions (dimensions) into a series of n frequency components ordered by their frequency. A subset of the resulting Fourier frequencies are used to approximate the original sequence in a lower dimensional space [17], and the tradeoff between analytical speed and accuracy can be varied according to the number of frequencies considered [45].

The DWT transforms data into the time-frequency domain, capturing both frequency and temporal location information [18,46,47], in contrast to DFT, which only provides frequency information. DWT is a set of averaging and differencing functions that may be used recursively to represent sequential data at different resolutions, and each resolution can be used as an approximation of the original data. Figure 1B depicts DWT transformations of a short nucleotide sequence.

In PAA, a numerical sequence is divided into n equally sized windows, the mean values of which together form a compressed sequence representation [20,21]. The selection of n determines the resolution of the compressed or approximate representation. While PAA is faster and easier to implement than the DFT and the DWT, unlike these two methods, PAA is irreversible, meaning that the original sequence cannot be recovered from the approximation. Figure 1C depicts an example of the PAA transformations of a short nucleotide sequence.

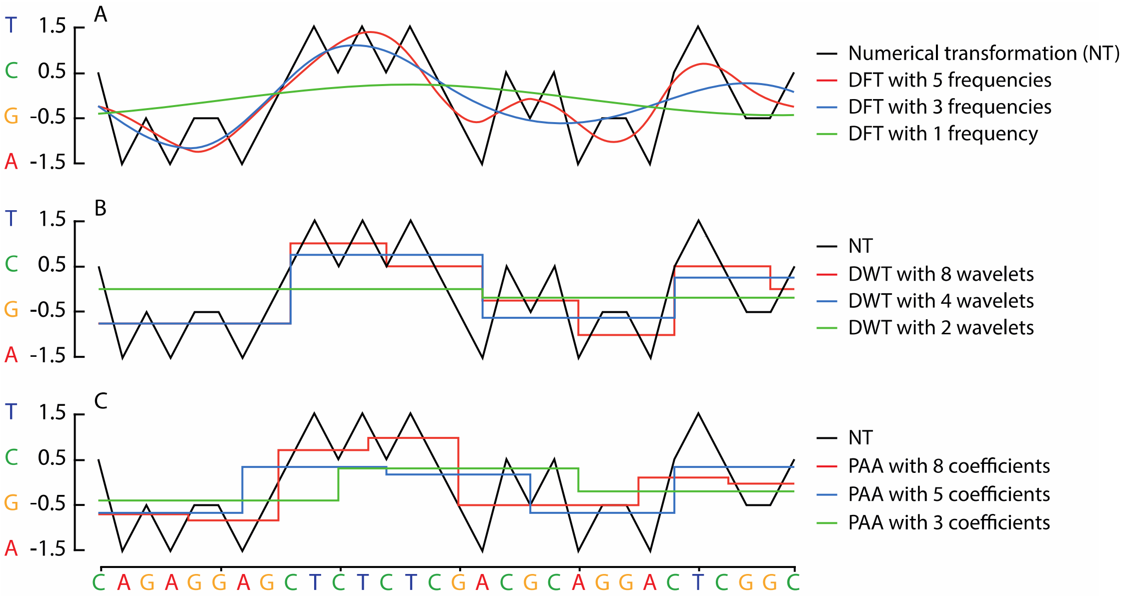

Figure 1. A numerically represented DNA sequence transformed at various levels of spatial resolution using the discrete Fourier transform (DFT) of the whole sequence (**A**), the Haar discrete wavelet transform (DWT) (**B**) and piecewise aggregate approximation (PAA) (**C**). A 30 nucleotide sequence (x-axis) is represented as a numerical sequence (black lines) using the real number representation method (y-axis where T = 1.5, C = 0.5, G = −0.5 and A = −1.5) for DFT approximations of the sequence with 5 (red), 3 (blue) and 1 (green) Fourier frequencies (**A**); DWT approximations of the same sequence with 8 level wavelets (red), 4 level wavelets (blue) and 2 level wavelets (green) (**B**); PAA approximations of the same sequence with 8 (red), 5 (blue) and 3 (green) coefficients (**C**).

2.3. Similarity Search Approaches for Sequential Data

Here, we adopt the Euclidian distance and *VP*-tree index to perform a fast *k*-nearest neighbour (*k*-NN) similarity search for aligning the reads to a reference genome.

In a *VP*-tree indexing structure, data is segregated using the distance between data points, thus implementing data partitioning in a metric space. A data point to use as a vantage point is selected (either randomly or by applying some heuristic to find and use the furthest point in the dataset [37]), and the rest of the data points are partitioned into two nodes based on their distance to that point. Data found to be closer to the vantage point than a given threshold (the median distance between all the data points and the vantage point) are assigned to the same node, and the rest of the data points to a different node. This function is repeated recursively in order to complete the partitioning process. The resulting indexing structure can then be used for fast identification of a *k*-nearest neighbour (*k*-NN) search. A *k*-NN-search returns the data points that are closest to a query *q*. Initially, the distance between the query *q* and the vantage point in the top node is calculated. If the distance between *q* and the vantage point satisfies a set of given conditions (the distance is smaller or larger than a given threshold – this threshold being the median distance between the vantage point and other data points within the node), a decision is made to visit either one or both of the child nodes. This process is repeated until the entire tree has been traversed. The *k* data points—in this case, reads—found closest to our query are the *k*-nearest neighbours to the query *q*.

2.4. Proposed Short Reads Processing Methodology

Our methodology for taxonomic classification, reference-based mapping and de novo assembly of short reads used time series and digital signal processing data transformation techniques. Figure 2 illustrates the fundamental concept of our approach. The short reads and reference genomes are mapped to a numerical space using an appropriate method from Table 1. Subsequently, lower dimensional approximations were generated for all data using the appropriate data transformation method, such as DFT, DWT and PAA. A *VP*-tree was constructed to allow fast data comparison. Depending on the application, the *VP*-tree was constructed either by using *k*-mer transformations obtained from the reference genomes or by using the short reads' transformations. Consequently, the best matches for our short reads' transformations were identified using a *k*-NN search approach on the *VP*-tree. As a final step, the results obtained from the *k*-NN search were re-evaluated in the original space to remove potential false positive results.

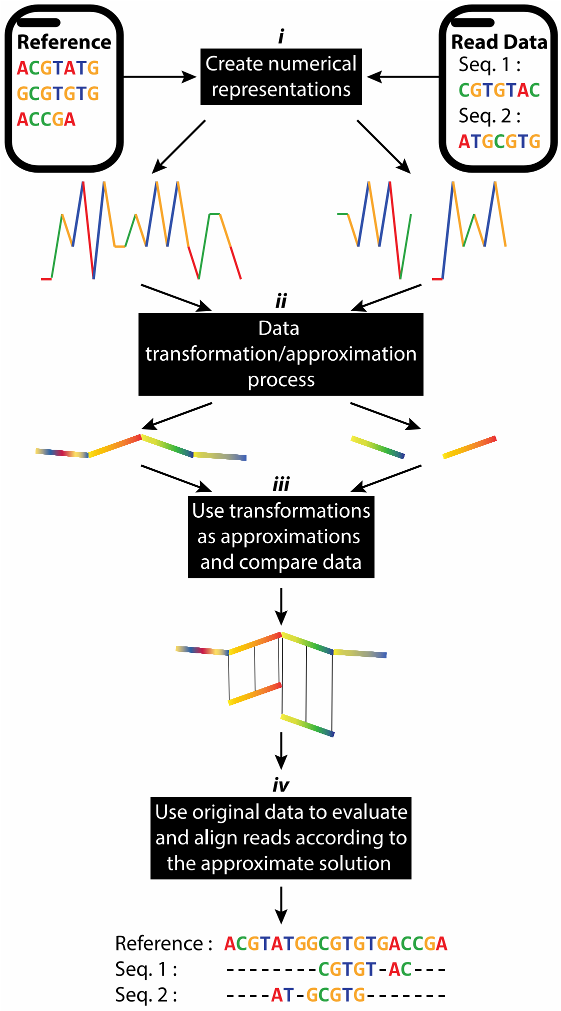
Reference
ACGTATG
GCGTGTG
ACCGA
i
Create numerical representations
Read Data
Seq. 1 :
CGTGTAC
Seq. 2 :
ATGCGTG
ii
Data transformation/approximation process
iii
Use transformations as approximations and compare data
iv
Use original data to evaluate and align reads according to the approximate solution
Reference : ACGTATGGCGTGTGACCGA
Seq. 1 : --------CGTGT-AC---
Seq. 2 : ----AT-GCGTG-------

Figure 2. Overview of our proposed methodology using time series transformation/approximation methods: (***i***) Creation of numerical representations of input sequences. (***ii***) Application of an appropriate signal decomposition method to transform sequences into their feature space. (***iii***) Use of approximated transformations to perform rapid data analysis in lower dimensional space. (***iv***) Validation of inferences against original, full-resolution input sequences. In the case of reference-based alignment and taxonomic classification, approximated read transformations were compared with a reference sequence. In our de novo implementation, pairwise comparisons were performed between all of the approximated read transformations.

2.5. Data

The implementations of our proposed methodologies were assessed with both simulated and real virus datasets. The simulated datasets were generated using CuReSim [48] and WGSIM (https://github.com/lh3/wgsim). Simulated data included information, such as the reference genome used, the alignment position and alignment direction, for each read, enabling rigorous evaluation of the proposed techniques. We used two simulators to examine our approach in a variety of use cases. CuReSim is highly customisable, allowing the user to control the type of variation (insertion, deletion and substitution) to simulate. WGSIM can simulate genomes with uniform insertion, deletion and substitution variation.

CuReSim was used to generate 16 HIV-1 HXB2 simulated datasets with different levels and types of variation. WGSIM was used to generate 4 mixed virus datasets with different levels of variation. Each simulation contained 200,000 reads generated using 5 Norovirus, 5 Ebola virus and 5 Respiratory syncytial virus (RSV) genomes, with various types and extents of simulated variation. HXB2 and simulated mixed virus datasets and corresponding reference genomes used to simulate them are deposited on GitHub (https://github.com/Avramis/Supporting-data/tree/master/Simulated%20Data). Table 2 contains detailed information about the simulated datasets.

Table 2. Simulated read data. Each row contains details for each simulated dataset (i.e., virus family, virus, GenBank ID, variation type, variation level, number of reads and simulator used to generate data). Abbreviations: Ins, insertions; Del, deletions and Sub, substitutions.

Family	**Virus**	**GenBank Genome ID**	**Variation Type (%)**			**Reads**	**Simulator**
			Ins	**Del**	**Sub**		
HIV	HXB2	K03455	0.0	0.0	0.0	2133	CuReSim
HIV	HXB2	K03455	0.0	0.0	1.0	2133	CuReSim
HIV	HXB2	K03455	0.0	0.0	2.0	2133	CuReSim
HIV	HXB2	K03455	0.0	0.0	3.0	2133	CuReSim
HIV	HXB2	K03455	0.0	0.0	4.0	2133	CuReSim
HIV	HXB2	K03455	0.0	0.0	5.0	2133	CuReSim
HIV	HXB2	K03455	0.5	0.5	0.0	2133	CuReSim
HIV	HXB2	K03455	1.0	1.0	0.0	2133	CuReSim
HIV	HXB2	K03455	1.5	1.5	0.0	2133	CuReSim
HIV	HXB2	K03455	2.0	2.0	0.0	2133	CuReSim
HIV	HXB2	K03455	2.5	2.5	0.0	2133	CuReSim
HIV	HXB2	K03455	0.5	0.5	1.0	2133	CuReSim
HIV	HXB2	K03455	1.0	1.0	2.0	2133	CuReSim
HIV	HXB2	K03455	1.5	1.5	3.0	2133	CuReSim
HIV	HXB2	K03455	2.0	2.0	4.0	2133	CuReSim
HIV	HXB2	K03455	2.5	2.5	5.0	2133	CuReSim
Mixed Viruses: Caliciviridae, Filoviridae, Pneumoviridae	Norovirus, Ebola virus, RSV	KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923, KP317922	0.0	0.0	0.0	200,000	WGSIM
Mixed Viruses: Caliciviridae, Filoviridae, Pneumoviridae	Norovirus, Ebola virus, RSV	KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923, KP317922	1.0	1.0	1.0	200,000	WGSIM
Mixed Viruses, Caliciviridae, Filoviridae, Pneumoviridae	Norovirus, Ebola virus, RSV	KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923, KP317922	3.33	3.33	3.33	100,000	WGSIM
Mixed Viruses, Caliciviridae, Filoviridae, Pneumoviridae	Norovirus, Ebola virus, RSV	KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923, KP317922	6.66	6.66	6.66	200,000	WGSIM

Furthermore, 15 publicly available real virus datasets were used for the evaluation of our methodology. The real datasets comprise 5 Norovirus, 5 Ebola virus and 5 human respiratory syncytial virus (RSV) short read datasets. Norovirus NGS datasets (ERR225628, ERR225629, ERR225631, ERR225632, ERR225633) were generated from diarrhoeal patients in Vietnam [49]. Group A rotavirus datasets were obtained from human and pig samples from Vietnam [50]. Human coronavirus NL63 datasets were obtained from Kenya [51]. The Ebola virus datasets (SRR3107337, SRR3107338, SRR3107340, SRR3107342, SRR3107343) were retrieved from the bioproject PRJNA309162, generated during the outbreaks in West Africa in 2013–2016 [52]. The human respiratory syncytial virus (RSV) datasets (ERR303259, ERR303260, ERR303261, ERR303262, ERR303263) [53] were generated from humans in Kenya. All 15 datasets are publicly available. The accession numbers of Sequence Read Archive (SRA) and European Nucleotide Archive (ENA) can be found in Table 3.

Table 3. Real short reads data. Rows contain information for each real reads' dataset (i.e., virus family, virus, genome strain GenBank ID, SRA project ID, number of reads and technology used to sequence data). SRA: Sequence Read Archive; ENA: European Nucleotide Archive.

Family	Virus	Amplicon/Random Primer	GenBank Genome ID	ENA/SRA_ID	Reads	Sequencing Technology
Caliciviridae	Norovirus	Amplicon	KM198486	ERR225628	2126502	Illumina MiSeq
Caliciviridae	Norovirus	Amplicon	KM198500	ERR225629	3037674	Illumina MiSeq
Caliciviridae	Norovirus	Amplicon	KM198511	ERR225631	3285078	Illumina MiSeq
Caliciviridae	Norovirus	Amplicon	KM198528	ERR225632	4361884	Illumina MiSeq
Caliciviridae	Norovirus	Amplicon	KM198529	ERR225633	5187234	Illumina MiSeq
Filoviridae	Ebola virus	Amplicon	KU296608	SRR3107337	522968	Ion Torrent PGM
Filoviridae	Ebola virus	Amplicon	KU296549	SRR3107338	771031	Ion Torrent PGM
Filoviridae	Ebola virus	Amplicon	KU296416	SRR3107340	186657	Ion Torrent PGM
Filoviridae	Ebola virus	Amplicon	KU296553	SRR3107342	478346	Ion Torrent PGM
Filoviridae	Ebola virus	Amplicon	KU296528	SRR3107343	42410	Ion Torrent PGM
Pneumoviridae	RSV	Amplicon	KP317934	ERR303259	7275032	Illumina MiSeq
Pneumoviridae	RSV	Amplicon	KP317922	ERR303260	9278070	Illumina MiSeq
Pneumoviridae	RSV	Amplicon	KP317946	ERR303261	11111114	Illumina MiSeq
Pneumoviridae	RSV	Amplicon	KP317923	ERR303262	13293226	Illumina MiSeq
Pneumoviridae	RSV	Amplicon	KP317952	ERR303263	15237848	Illumina MiSeq

The HIV-1 HXB2 genome (K03455) was used as a reference index to align and/or run the taxonomic classification analysis for the HXB2 simulated dataset. The Norovirus genome KM198509, the Ebola virus genome KM034562 and the RSV genome KP317934 were used as a reference index to align and/or run the taxonomic classification analysis for the mixed virus datasets. The Norovirus genome KM198509 was used to run the taxonomic classification analysis on the real Norovirus datasets, the Ebola virus genome KM034562 was used to run the taxonomic classification analysis on the real Ebola datasets and the RSV genome KP317934 was used to perform the taxonomic classification analysis on the real RSV datasets. All reference genomes used in this study are available from the NCBI (https://www.ncbi.nlm.nih.gov/genome), and accession numbers can be found in Table 4.

Table 4. Reference genomes used during classification and reference-based alignment.

Family	Virus	GenBank ID:	Length (nt)
Retroviridae	Human immunodeficiency virus 1 (HXB2)	K03455	9179
Caliciviridae	Norovirus	KM198509.1	7425
Filoviridae	Zaire ebolavirus	KM034562.1	18957
Pneumoviridae	Human orthopneumovirus (Respiratory Syncytial Virus)	KP317934.1	15233

2.6. Classification and Alignment Evaluation

The accuracy of a classification and an alignment tool can be quantified in terms of the *F*-measure [48], a balanced measure of precision and recall, with precision = true positive/true positive + false positive, recall = true positive/true positive + false negative and the *F*-measure = 2 × (precision *x* recall)/(precision + recall) [48]. In the case of simulated data, information concerning the position of the read on the reference and alignment direction can be used to establish the correctness of alignment, and thereby provide a more informative *F*-measure score. Unclassified reads are considered a false negative result. Any reported match to the correct region of the genome in the correct direction is considered a true positive result. However, if the alignment position or direction information is unavailable, the *F*-measure can be calculated from the number of hits reported for a read, or the absence of a hit. Again, unclassified reads are considered false negative results, and classified reads are considered true positive results. In the case of mixed genome data, the *F*-measure score can be calculated by taking into consideration the number of hits that are reported for a read, as well as if a read is assigned to a reference genome from the same family. If a read is assigned to a genome from a different virus family, it is considered a false positive result, while unclassified reads are considered a false negative result.

3. Results

3.1. Classification by Numbers (CBN)

For the taxonomic classification analysis, a classification tool was implemented in C++ (https://github.com/Avramis/ClassificationByNumbers). The implementation was developed to evaluate our methodology but was not optimised for speed. Users might specify parameters, such as the representation method, transformation method, search stringency and the *k*-mer length. A *VP*-tree indexing structure classified reads using a given set of genomic references. *VP*-tree construction began with the extraction of all unique *k*-mers, of a user-specified length *k*, from the set of supplied reference genomes. Each unique *k*-mer was represented in numerical sequence and then transformed into a lower dimensional space. The transformed data were then used to generate the *VP*-tree indexing structure. Subsequently, each short read from a query set was converted into numerical space, transformed to a lower dimensional space and evaluated against the *VP*-tree. The approximate solution arising from this was then evaluated using the original data to identify false positive matches. The CBN algorithm generated two output files. The first output was a text file providing detailed information on all of the classification matches generated for each read, including the reference name, the direction in which the query read was aligned to the reference, the start and end position of the query on the reference, the alignment score, the CIGAR string describing how the read aligns with the reference and the actual alignment of the query read on the reference genome. The second tabular output file provided a brief overview of the alignment. Each line contained the name of the read, the number of classifications generated for that particular read, the highest classification score obtained, the name of the reference, which provided the highest classification score, the alignment direction and starting position on the reference.

The CBN tool was evaluated against NCBI-BLAST 2.8.1 BLASTn [54] and Kaiju 1.6.3 [55] classifier tools. BLASTn performs the analysis in nucleotide space, whereas Kaiju translates nucleotide sequences from every possible reading frame and performs the analysis in protein sequence space. Figures 3–5 illustrate the results of the classification evaluation process. Both BLASTn and Kaiju were evaluated using their default parameters. CBN was evaluated using *k*-mers of 100, 150, 200, 250 and 300 for the HXB2 simulated reads and 50, 100 and 150 for the mixed virus and real datasets. For the DFT and PAA methods, we evaluated the use of transformation/approximations with 2, 4, 6, 8, 10 and 12 Fourier frequencies or PAA coefficients, respectively. For the DWT variant, we tested the cases of 2, 4, 8, 16 and 32 wavelets.

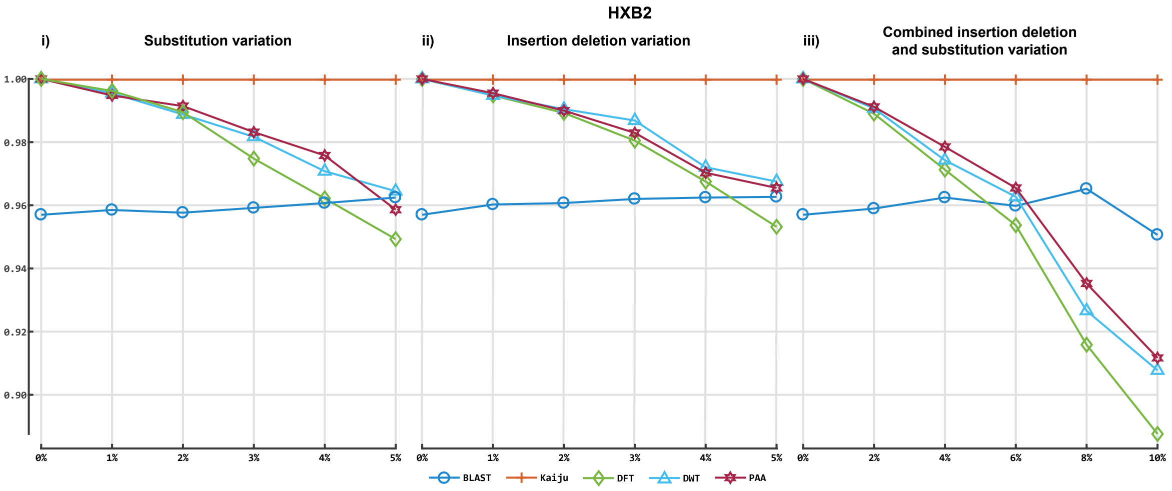

Figure 3. Accuracy of our prototype classification implementation and two established tools on HIV-1 HXB2 simulated datasets. All plots illustrate the *F*-measures obtained on the 16 different HIV datasets. The y-axis indicates the *F*-measure score, and the x-axis depicts the reads data files. Plot 3-i depicts the *F*-measures obtained for each classifier on the simulations with 0% to 5% of substitution variation rate. Plot 3-ii illustrates the *F*-measures obtained for each classifier on the simulations with 0% to 5% uniform insertion/deletion variation, and plot 3-iii illustrates the *F*-measures obtained for each tool on simulations of uniform 0% to 10% insertion/deletion and substitution variation.

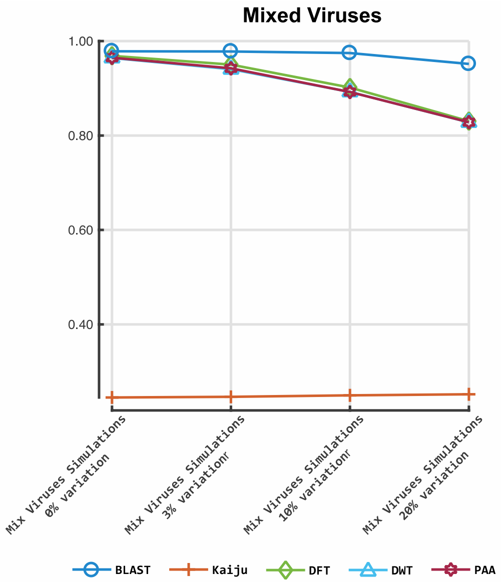

Figure 4. Accuracy of our prototype classification implementation and two established tools on mixed viruses simulated datasets. The *y*-axis indicates the *F*-measure score, and the *x*-axis depicts the reads data files. The plot depicts the *F*-measures obtained for each classifier on the mixed virus simulations. DFT: discrete Fourier transform; DWT: discrete wavelet transform; PAA: piece-wise aggregate approximation.

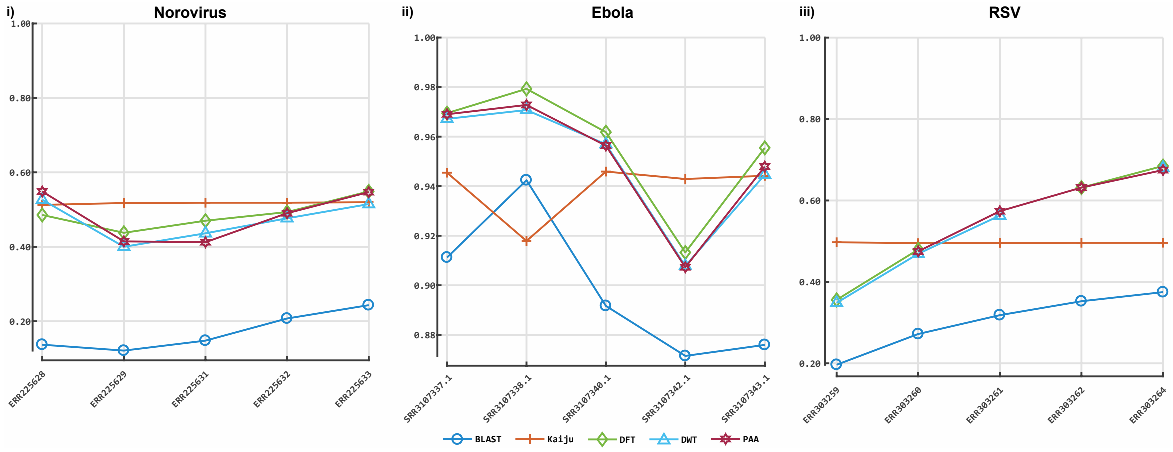

Figure 5. Accuracy of our prototype classification implementation and two established tools on real sequences. The y-axis indicates the F-measure score, and the x-axis depicts the reads data files. Plot 5-i depicts the F-measures obtained for each classifier on the Norovirus sequences data. Plot 5-ii illustrates the F-measures obtained for each classifier on the Ebola sequence data. Plot 5-iii illustrates the F-measures obtained for each tool on Respiratory syncytial virus (RSV) sequence data. DFT: discrete Fourier transform; DWT: discrete wavelet transform; PAA: piece-wise aggregate approximation.

Figure 3 shows the results obtained from the classification process on HIV- 1 HXB2 data. Figure 4 illustrates the results of the mixed virus datasets. Figure 5 illustrates the results obtained from the real data. For taxonomic classification of HIV-1 HXB2 simulated reads, where the short reads were classified against the genome used to generate them, Kaiju reported the highest accuracy scores. CBN outperformed BLASTn in most cases, falling behind in terms of accuracy only on datasets with high variation rates. For the mixed virus simulated datasets, where reads were classified against species strains related to those used to generate reads, BLASTn correctly assigned the most species, followed closely by CBN and finally Kaiju. In the evaluation of the tools on the real data, where reads were classified using a publicly available species-specific reference sequence, CBN generated more accurate results than other tools, followed by Kaiju and BLASTn.

3.2. Alignment by Numbers (ALBN)

To test the applicability of sequential data transformations and feature selection for read alignment, we implemented a prototype k-NN read aligner (Figure 6) in C++ (available at https://github.com/Avramis/Alignment_by_numbers). As with the CBN classification analysis, the ALBN code was not optimised for speed. Users might specify parameters, such as the representation method, transformation method, search stringency and the k-mer length used for seeding alignments. The algorithm's output was used to construct gapped alignments in the widely used Sequence Alignment/Map (SAM) file format.

```
1) Represent short reads and reference genome as numerical sequences.
2) Select a k-mer length.
3) Create transformations of each reference sequence k-mer, build VP-tree, and create
transformations of the initial k-mer of each short read.
4) Identify candidate alignments using data transformations.
for each read i
    candidate_alignments[i] = VPtree.k-NNSearch(query i)
end
5) Align approximate results with original data using the Smith-Waterman (SW)
algorithm:
for each read i
    best_score = null
   best_aln = []
    for each k neighbour in candidate_alignments[i]
        if SW_score(k neighbour, read i)
            best_score = SW_score(k neighbour, read i)
            best_aln = SW_aln(k neighbour, read i)
        end
    end
end
6) Output alignment in Sequence Alignment/Map (SAM) format.
```

Figure 6. Pseudocode for the alignment procedure.

The ALBN tool was evaluated against a set of well-established, widely used, state-of-the-art tools, such as Bowtie2 (version 2.3.1) [56], BWA-MEM (version 0.7.16) [7], GraphMap (version 0.5.2) [57] and Segmehl (version 0.3.4) [58]. Existing state-of-the-art tools were evaluated with default settings. ALBN was evaluated using *k*-mer lengths of 100, 150, 200, 250 and 300 for the HXB2 simulated reads and 50, 100 and 150 for the mixed virus and real datasets. For the DFT and PAA variants, we evaluated the use of transformation/approximations with 2, 4, 6, 8, 10 and 12 frequencies and PAA coefficients accordingly. For the DWT variant, we tested the cases of 2, 4, 8, 16 and 32 wavelets.

Each aligner's accuracy was quantified in terms of the *F*-measure [48]. CuReSim provides information, such as the simulated read's origin on the reference genome and its alignment direction, enabling evaluation of each aligner's output and calculation of alignment accuracy in terms of the *F*-measure. For mixed virus datasets, tool performance was evaluated in terms of ability to match and align reads to the appropriate virus reference genome. For the real data, *F*-measures were calculated according to the number of reads aligned to the given genome or otherwise.

Figures 7–9 illustrate the *F*-measures obtained by evaluating alignments from each aligner. Figure 7 illustrates alignment performance for each of the 16 datasets simulated using the K03455 HIV-1 HXB2 reference genome. Figure 8 illustrates the alignment performance for virus reads simulated with Norovirus genome KM198509.1, Ebola genome KM034562.1 and the RSV genome KP317934.1. Figure 9i–iii illustrate alignment performance (*F*-measure) for alignments of real Norovirus, Ebola virus and RSV sequences against the same reference genomes as those used for simulation.

ALBN provided accurate results in all scenarios tested. Regarding the HIV-1 HXB2 data, where short reads were aligned to the genome used to generate them, ALBN provided the most accurate results in all 16 cases, followed by Bowtie2. This was also the case for the mixed virus datasets, where reads were aligned to reference strains related to those used to generate the dataset. In both cases, GraphMap and BWA-MEM were third and fourth in terms of accuracy, respectively. ALBN also generated the most accurate alignment results using real data, where reads were aligned to species-specific reference genomes.

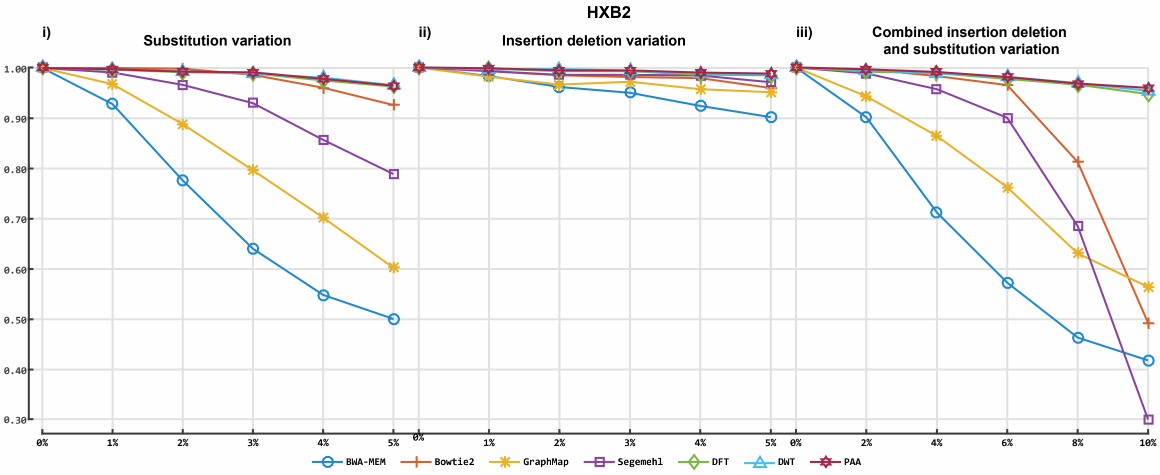

Figure 7. Accuracy of our prototype reference alignment implementation and four established tools on HIV-1 HXB2 simulated datasets. This Figure illustrates the *F*-measures obtained on the 16 different HIV datasets. Plot 6-(**i**) depicts the *F*-measures obtained for each aligner on the simulations with 0% to 5% of substitution variation rate. Plot 6-(**ii**) illustrates the *F*-measures obtained for each aligner on the simulations with 0% to 5% uniform insertion/deletion variation, and plot 6-(**iii**) illustrates the *F*-measures obtained for each tool on simulations of uniform 0% to 10% insertion/deletion and substitution variation. DFT: discrete Fourier transform; DWT: discrete wavelet transform; PAA: piece-wise aggregate approximation.

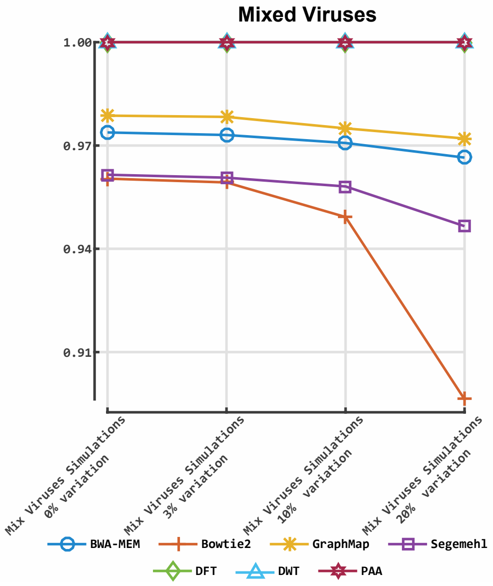

Figure 8. Accuracy of our prototype aligner implementation and four established tools on mixed viruses simulated datasets. The *y*-axis indicates the *F*-measure score, and the *x*-axis depicts the reads data files. The plot depicts the *F*-measures obtained for each aligner on the mixed virus simulations. DFT: discrete Fourier transform; DWT: discrete wavelet transform; PAA: piece-wise aggregate approximation.

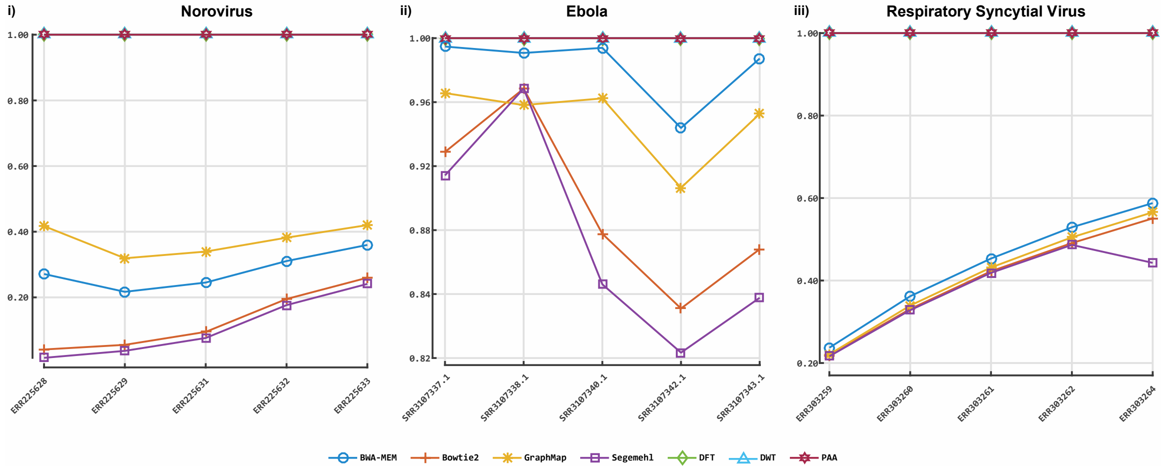

Figure 9. Accuracy of our prototype aligner implementation and four established tools on real sequences datasets. The *y*-axis indicates the *F*-measure score, and the *x*-axis depicts the reads data files. Plot 8-(**i**) depicts the *F*-measures obtained for each aligner on the Norovirus sequences data. Plot 8-(**ii**) illustrates the *F*-measures obtained for each aligner on the Ebola sequences data. Plot 8-(**iii**) illustrates the *F*-measures obtained for each tool on the Respiratory syncytial virus (RSV) sequences data. DFT: discrete Fourier transform; DWT: discrete wavelet transform; PAA: piece-wise aggregate approximation.

3.3. *De novo Assembly by Numbers*

Lastly, to test the applicability of this approach to the de novo assembly of short reads, we implemented assembly by numbers (ASBN), a prototype algorithm for all-against-all *k*-mer comparison, using data transformations/approximation. Note, preliminary results have been presented as a conference paper [59]. Figure 10 illustrates the main concept of our de novo assembly approach. For the ASBN tool, reads are represented as numerical sequences using an appropriate numerical representation method (Table 1). Here, we used the tetrahedron numerical representation. Every *k*-mer of each numerically represented read was identified and transformed to lower dimensional space using the chosen transformation method. All *k*-mers' transformations were used to build a *VP*-tree, to allow for fast data comparison. Afterwards, all *k*-mers were compared to the rest of the data using the *VP*-tree index. Information from the data comparison was used to construct a weighted graph similar to that shown in Figure 10A. The shortest path on the weighted graph was identified with a breadth-first search (BFS) (Figure 10B). Reads overlaps were used to generate an OLC alignment of short reads (Figure 10C).

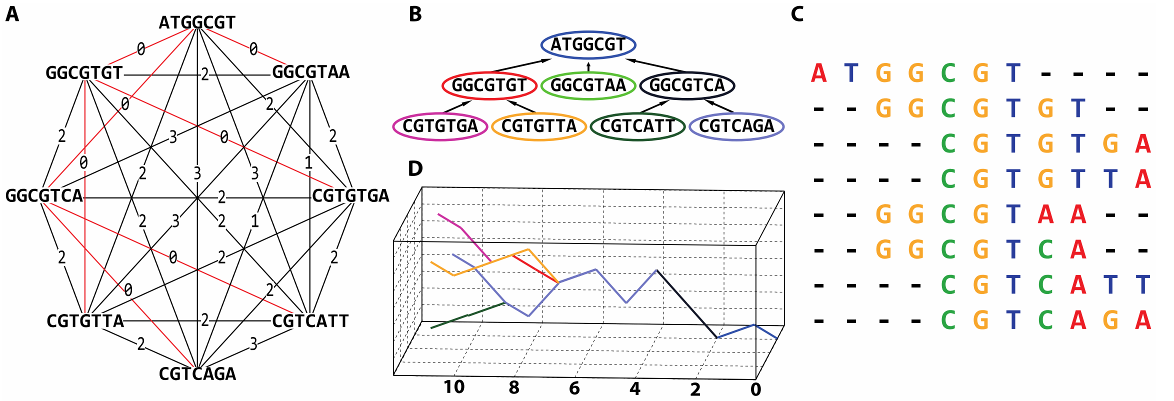

Figure 10. A de novo assembly methodology for numerically represented nucleotide reads. All-against-all sequence comparison (**A**) enables the construction of a read graph with weighted edges. The weight assigned to each edge is the smallest pairwise distance obtained between every possible *k*-mer representation of the two reads. In this example, a 5-mer was used. The smallest distance between every possible *k*-mer can be obtained by either using a sliding window approach or break reads every possible subsequence with length *k*. (**B**) The shortest path in the graph is identified with a breadth-first search algorithm (red coloured edges) thereby (**C**) enabling read alignment. A DNA walk representation of the overlapped reads (**D**) may subsequently be used as a three-dimensional graphical portrayal of the reads, illustrating alignment characteristics.

The ASBN assembler was compared with Megahit (version 1.1.3) [60] and SPAdes (version 3.13.0) [61] de novo assemblers on the HIV-1 HXB2, and mixed virus simulated datasets accordingly. Megahit, SPAdes, BLASTn and Kaiju were evaluated using default parameters. ASBN was evaluated using *k*-mer lengths 100, 150, 200, 250 and 300 for the HXB2 simulated reads and 50, 100 and 150 for the mixed virus datasets. For the DFT and PAA variants, we evaluated the use of transformation/approximations with 2, 4, 6, 8, 10 and 12 frequencies and PAA coefficients accordingly. For the DWT variant, we tested the cases of 2, 4, 8, 16 and 32 wavelets.

The derived contigs from each assembler were evaluated against the reference genomes used to generate the data simulations with BLASTn [54]. From the BLASTn output, information about the contigs' alignment position on the genome and the length of the alignment were obtained. Subsequently, a measure of assembly contiguity and the sum of gaps/mismatches were calculated and plotted on an X-Y matrix (similar to Figures 11 and 12) with x being the total coverage of the genomes generated and y being the total number of gaps in the coverage. A perfect assembly would have x = full genome length and $y = 0$, indicating that the contig is identical to the genome in terms of length and nucleotide composition. For the HIV-1 HXB2 datasets, the contigs were evaluated against the K03455 genome, and the contigs obtained from the mixed virus datasets were evaluated against the 15 different genomes: KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923 and KP317922.

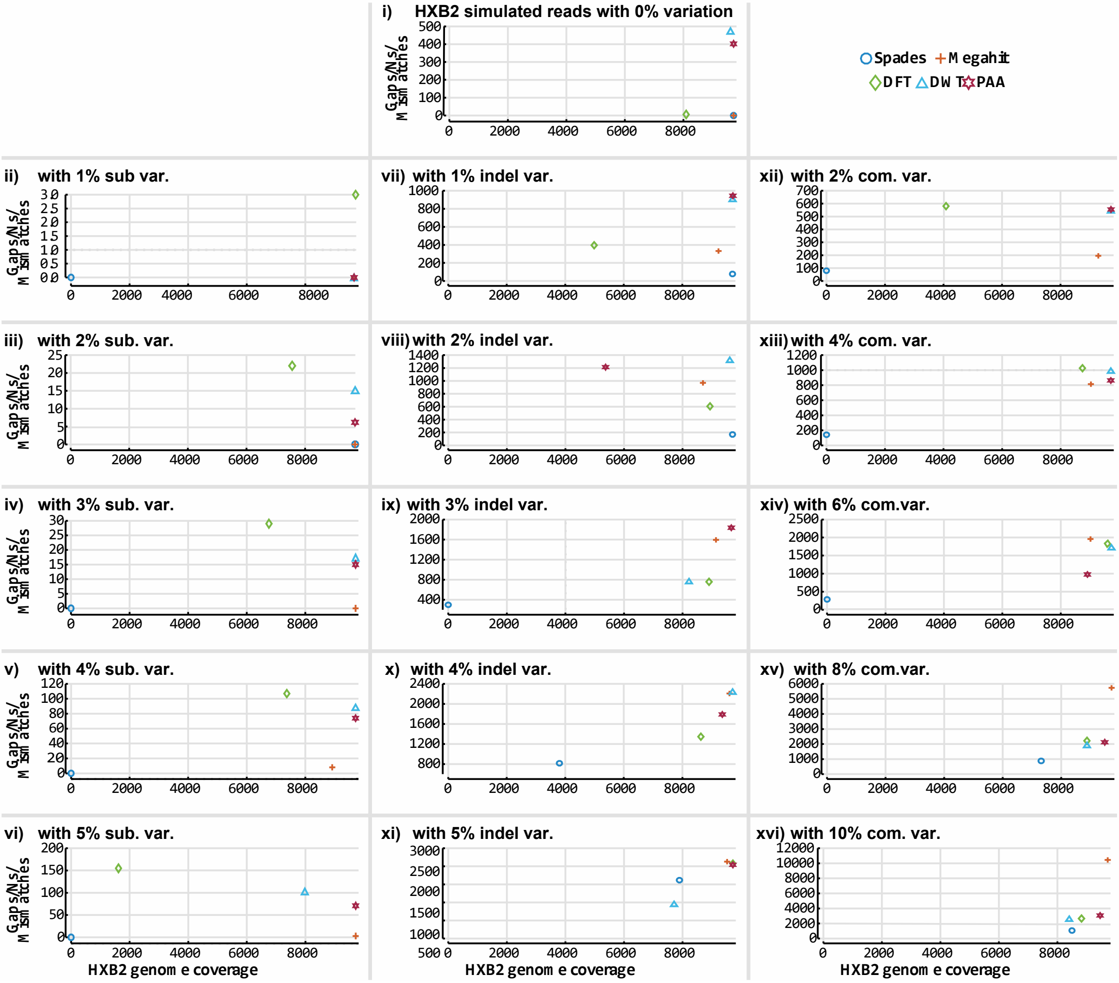

Figure 11. Accuracy of our prototype de novo assembly implementation and two established tools on HIV-1 HXB2 simulated datasets. The contigs obtained for each assembler were evaluated against the reference genome used to generate the simulated data. BLASTn was used to align all contigs to an HIV-1 HXB2 reference genome and determine genome coverage. The *y*-axis indicates the number of gaps and mismatches that exist in the contigs obtained for each tool, and the *x*-axis depicts the length of the genome the reported contigs cover. The contigs obtained from the assembly of the HIV-1 HXB2 simulated short read data were evaluated against the K03455 reference genome. Plot 10-**i** illustrates results obtained from all assemblers on variation-free data. Plots 10-**ii** to 10-**vi** illustrate results obtained from all assemblers on data with different levels of substitution variation. Plots 10-**vii** to 10-**xi** illustrate results obtained from all assemblers on data with different levels of insertion/deletion variation. Plots 10-**xii** to 10-**xvi** illustrate results obtained from all assemblers on data with different levels of combined insertion/deletion and substitution variation.

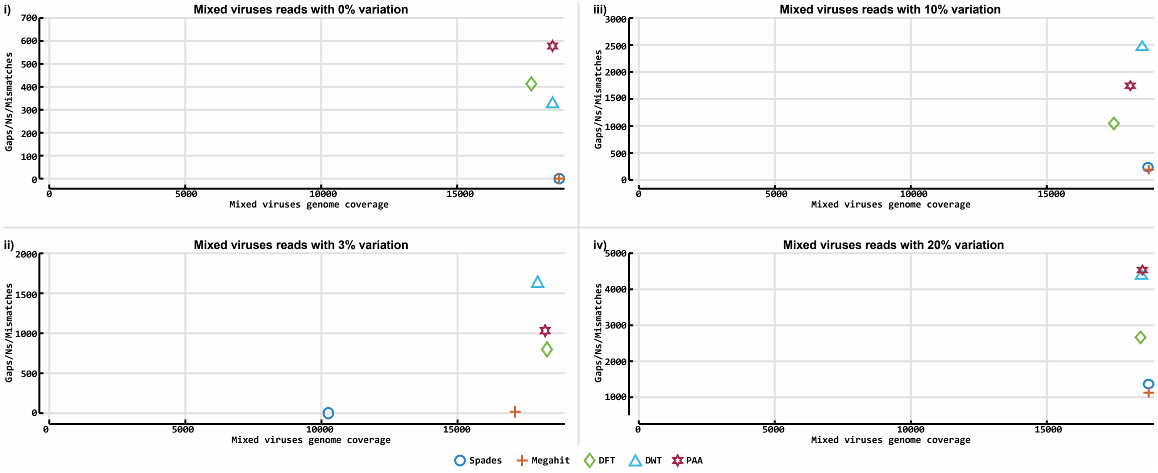

Figure 12. Accuracy of our prototype de novo assembly implementation and two established tools on mixed viruses simulated datasets. The contigs obtained for each assembler were evaluated against the reference genome that was used to generate the simulated data. BLASTn was used to align all contigs to an HIV-1 HXB2 reference genome and determine how much of the particular genome they cover. The *y*-axis indicates the number of gaps and mismatches that exist in the contigs obtained for each tool, and the *x*-axis depicts the length of the genome the reported contigs cover. The contigs obtained from the mixed virus simulated dataset were evaluated against the KM198529, KM198528, KM198511, KM198500, KM198486, KU296608, KU296553, KU296549, KU296528, KU296416, KP317952, KP317946, KP317934, KP317923 and KP317922 references genomes. Plots 11-**i** to 11-**iv** illustrate results obtained from all assemblers on data with 0%, 3%, 10% and 20% variation levels accordingly.

Figure 11 illustrates the assembly results of SPAdes, Megahit and all three variants of ASBN on the 16 simulated HIV-1 HXB2 datasets. Figure 12 illustrates the assembly results on the mixed virus simulated databases. Although ASBN processes data and assembles short reads in a lower dimensional space, it nevertheless generated contigs that collectively cover the expected genome length and provided comparable results to both existing state of the art de novo assemblers tested in this experiment (Figures 11 and 12). In all cases, ASBN generated contigs spanning the whole genomes of their respective viral species.

4. Discussion

Although well-established data compression methods for reversible compression of one-dimensional and multivariate signals, images, text and binary exist [62–64], there are very few examples of their application to biological sequence data. We have developed algorithms incorporating signal compression methods for three common biological sequence analysis problems: classification, alignment and de novo assembly of NGS short read virus data. Our results in Figures 3–12 show that this approach permits accurate classification of de novo assembly and reference alignment in spite of high rates of sequence variation or the use of a divergent reference genome. Data approximation/summarisation techniques, such as the DFT, the DWT and the PAA, can be used to extract major features of sequence data and to suppress noise or low-level variation. This allows sequence comparison exploiting the major characteristics of the data, thus enabling the identification of similarities that might otherwise be concealed by minor variation or sequencing error/noise.

Collectively, our results demonstrate that complete nucleotide-level sequence resolution is not a prerequisite of accurate sequence analysis and that analytical performance can be preserved and even enhanced through appropriate dimensionality reduction (compression) of sequences. While our implementations use *k*-mers, the nature of the transformation/compression methods used shows that optimal *k*-mer length selection is far less important than the conventional exact *k*-mer matching methods. The inherent error tolerance of the approach also permits the use of longer *k*-mers than typically used in conventional sequence comparison algorithms, reducing the computational burden of pairwise comparison, and thus, in de novo assembly specifically, the complexity of building and searching an assembly graph.

Efficient mining of terabase-scale biological sequence datasets requires looking beyond substring-indexing algorithms [65] towards more versatile methods of compression for both data storage and analysis. The use of probabilistic data structures can considerably reduce the computer memory required for in-memory sequence lookups at the expense of a few false positives, and Bloom filters and related data structures have seen broad application in *k*-mer centric tasks, such as error correction [66], in silico read normalisation [67] and de novo assembly [68,69]. However, while these hash-based approaches perform well on datasets with high sequence redundancy, for large datasets with many distinct *k*-mers, large amounts of memory are still necessary [67]. Lower bounding transformations and approximation methods (such as the DFT, the DWT and PAA) can exhibit the same attractive one-sided error offered by these probabilistic data structures, but instead of hash tables, use concrete and reusable sequence representations.

Furthermore, transformations allow compression of standalone sequence composition, enabling flexible reduction of sequence resolution according to analytical requirements, so that redundant sequence precision need not hinder analysis. While the problem of read alignment to a known reference sequence is largely considered solved, assembly of large genomes remains a formidable problem in computing. Moreover, consideration of the metagenomic composition of mixed biological samples, as demonstrated, further extends the scope and scale of the assembly problem beyond what is tractable using conventional sequence comparison approaches. By implementing a reference-based aligner and de novo assembler, we have demonstrated that using compressed numerical representations offers a versatile approach for reconstructing genomes and metagenomes sequenced with short reads.

Emerging long read sequencing technologies bring new challenges for sequence data analysis. Whilst the error rate of Oxford Nanopore sequencing platform, for example, has decreased considerably since the technology's introduction [70,71], the relatively high error rate still limits the scope of downstream analyses [72]. Efficient algorithmic approaches are needed to (1) identify sequence identity/infer homology in spite of abundant insertion/deletion errors associated with the platform, which are problematic for approaches dependent on exact subsequence matching and (2) to overcome issues relating to high data dimensionality and the curse of dimensionality [73]. Both in terms of the raw electric current traces generated by DNA translocation through a nanopore and the corresponding base-called sequences, the resemblance between long reads and time series data from other fields is striking, such that the various transformations/approximations we have implemented will be directly applicable.

In conclusion, nucleotide sequences may be effectively represented as numerical series, enabling the application of existing analytical methods from a variety of mathematical and engineering fields for the purposes of sequence alignment and assembly. By applying established signal decomposition methods, compressed representations of nucleotide sequences can be created, permitting reductions in the spatiotemporal complexity of their analysis, without necessarily compromising analytical accuracy.

Author Contributions: A.T. designed and wrote the methods and software, and performed the data analysis with the help from B.C., M.V.T.P., S.K. and M.C., B.C. and M.C. generated the simulated data, and B.C., M.V.T.P., and M.C. helped perform the data evaluation. A.T. and D.L.R. conceived the study. A.T. and B.C. wrote the manuscript with comments from M.V.T.P., M.C. and D.L.R. All authors read and approved the final manuscript.

Funding: This work has been supported by the Welcome Trust [097820/Z/11/B], the BBSRC [BB/H012419/1, BB/M001121/1 and BC by a BBSRC DTP studentship] and the VIROGENESIS project, which receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 634650.

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

References

1. Margulies, M.; Egholm, M.; Altman, W.E.; Attiya, S.; Bader, J.S.; Bemben, L.A.; Berka, J.; Braverman, M.S.; Chen, Y.-J.; Chen, Z. Genome sequencing in microfabricated high-density picolitre reactors. *Nature* **2005**, *437*, 376–380. [CrossRef] [PubMed]
2. Bentley, D.R.; Balasubramanian, S.; Swerdlow, H.P.; Smith, G.P.; Milton, J.; Brown, C.G.; Hall, K.P.; Evers, D.J.; Barnes, C.L.; Bignell, H.R. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature* **2008**, *456*, 53–59. [CrossRef]
3. Rothberg, J.M.; Hinz, W.; Rearick, T.M.; Schultz, J.; Mileski, W.; Davey, M.; Leamon, J.H.; Johnson, K.; Milgrew, M.J.; Edwards, M. An integrated semiconductor device enabling non-optical genome sequencing. *Nature* **2011**, *475*, 348–352. [CrossRef]
4. Eid, J.; Fehr, A.; Gray, J.; Luong, K.; Lyle, J.; Otto, G.; Peluso, P.; Rank, D.; Baybayan, P.; Bettman, B. Real-time DNA sequencing from single polymerase molecules. *Science* **2009**, *323*, 133–138. [CrossRef]
5. Salipante, S.J.; Roach, D.J.; Kitzman, J.O.; Snyder, M.W.; Stackhouse, B.; Butler-Wu, S.M.; Lee, C.; Cookson, B.T.; Shendure, J. Large-scale genomic sequencing of extraintestinal pathogenic Escherichia coli strains. *Genome Res.* **2015**, *25*, 119–128. [CrossRef] [PubMed]
6. Rose, R.; Constantinides, B.; Tapinos, A.; Robertson, D.L.; Prosperi, M. Challenges in the analysis of viral metagenomes. *Virus Evol.* **2016**, *2*. [CrossRef] [PubMed]
7. Li, H.; Durbin, R. Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics* **2009**, *25*, 1754–1760. [CrossRef] [PubMed]
8. Shrestha, A.M.S.; Frith, M.C.; Horton, P. A bioinformatician's guide to the forefront of suffix array construction algorithms. *Brief. Bioinform.* **2014**, *15*, 138–154. [CrossRef]
9. Myers, E.W. Toward simplifying and accurately formulating fragment assembly. *J. Comput. Biol.* **1995**, *2*, 275–290. [CrossRef]
10. Kececioglu, J.D.; Myers, E.W. Combinatorial algorithms for DNA sequence assembly. *Algorithmica* **1995**, *13*, 7–51. [CrossRef]
11. Earl, D.; Bradnam, K.; John, J.S.; Darling, A.; Lin, D.; Fass, J.; Yu, H.O.K.; Buffalo, V.; Zerbino, D.R.; Diekhans, M. Assemblathon 1: A competitive assessment of de novo short read assembly methods. *Genome Res.* **2011**, *21*, 2224–2241. [CrossRef] [PubMed]
12. Iqbal, Z.; Caccamo, M.; Turner, I.; Flicek, P.; McVean, G. De novo assembly and genotyping of variants using colored de Bruijn graphs. *Nat. Genet.* **2012**, *44*, 226–232. [CrossRef]
13. Pevzner, P.A.; Tang, H.; Waterman, M.S. An Eulerian path approach to DNA fragment assembly. *Proc. Natl. Acad. Sci. USA* **2001**, *98*, 9748–9753. [CrossRef]
14. Bradnam, K.R.; Fass, J.N.; Alexandrov, A.; Baranay, P.; Bechner, M.; Birol, I.; Boisvert, S.; Chapman, J.A.; Chapuis, G.; Chikhi, R. Assemblathon 2: Evaluating de novo methods of genome assembly in three vertebrate species. *Gigascience* **2013**, *2*, 1–31. [CrossRef] [PubMed]
15. Archer, J.; Rambaut, A.; Taillon, B.E.; Harrigan, P.R.; Lewis, M.; Robertson, D.L. The evolutionary analysis of emerging low frequency HIV-1 CXCR4 using variants through time—An ultra-deep approach. *PLoS Comput. Biol.* **2010**, *6*, e1001022. [CrossRef] [PubMed]
16. Clement, N.L.; Thompson, L.P.; Miranker, D.P. ADaM: Augmenting existing approximate fast matching algorithms with efficient and exact range queries. *BMC Bioinform.* **2014**, *15*, S1. [CrossRef] [PubMed]
17. Agrawal, R.; Faloutsos, C.; Swami, A. Efficient similarity search in sequence databases. In Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms, Chicago, IL, USA, 13–15 October 1993.
18. Chan, K.-P.; Fu, A.-C. Efficient time series matching by wavelets. In Proceedings of the 15th International Conference on Data Engineering, Sydney, Australia, 23–26 March 1999; pp. 126–133.
19. Woodward, A.M.; Rowland, J.J.; Kell, D.B. Fast automatic registration of images using the phase of a complex wavelet transform: Application to proteome gels. *Analyst* **2004**, *129*, 542–552. [CrossRef] [PubMed]
20. Geurts, P. Pattern extraction for time series classification. In Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery, Freiburg, Germany, 3–7 September 2001; pp. 115–127.
21. Keogh, E.; Chakrabarti, K.; Pazzani, M.; Mehrotra, S. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM SIGMOD Record* **2001**, *30*, 151–162. [CrossRef]

22. Shumway, R.H.; Stoffer, D.S.; Stoffer, D.S. *Time Series Analysis and Its Applications with R examples*, 2nd ed.; Springer: New York, NY, USA, 2006.
23. Silverman, B.; Linsker, R. A measure of DNA periodicity. *J. Theor. Biol.* **1986**, *118*, 295–300. [CrossRef]
24. Cheever, E.; Searls, D.; Karunaratne, W.; Overton, G. Using signal processing techniques for DNA sequence comparison. In Proceedings of the Fifteenth Annual Northeast Bioengineering Conference, Boston, MA, USA, 27–28 March 1989; pp. 173–174.
25. Katoh, K.; Misawa, K.; Kuma, K.i.; Miyata, T. MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.* **2002**, *30*, 3059–3066. [CrossRef] [PubMed]
26. Kwan, H.K.; Arniker, S.B. Numerical representation of DNA sequences. In Proceedings of the 2009 IEEE International Conference on Electro/Information Technology, Windsor, ON, Canada, 7–9 June 2009; pp. 307–310.
27. Yi, B.-K.; Faloutsos, C. Fast time sequence indexing for arbitrary Lp norms. In Proceedings of the 26th roceedings of 26th International Conference on Very Large Data Bases, Cairo, Egypt, 10–14 September 2000; pp. 385–394.
28. Keogh, E.; Ratanamahatana, C.A. Exact indexing of dynamic time warping. *Knowl. Inf. Syst.* **2005**, *7*, 358–386. [CrossRef]
29. Vlachos, M.; Kollios, G.; Gunopulos, D. Discovering similar multidimensional trajectories. In Proceedings of the 18th International Conference on Data Engineering, San Jose, CA, USA, 26 February–1 March 2002; pp. 673–684.
30. Kotsakos, D.; Trajcevski, G.; Gunopulos, D.; Aggarwal, C.C. *In Data Clustering: Algorithms and Applications*; Aggarwal, C.C., Reddy, C., Eds.; CRC Press: Boca Raton, FL, USA, 2013; Chapter 15; pp. 357–379.
31. Chávez, E.; Navarro, G.; Baeza-Yates, R.; Marroquín, J.L. Searching in metric spaces. *ACM Comput. Surv. (CSUR)* **2001**, *33*, 273–321. [CrossRef]
32. Beckmann, N.; Kriegel, H.-P.; Schneider, R.; Seeger, B. The R*-tree: An efficient and robust access method for points and rectangles. *SIGMOD Rec.* **1990**, *19*, 322–331. [CrossRef]
33. Agrawal, R.; Lin, K.; Sawhney, H.S.; Shim, K. Fast similarity search in the presence of noise, scaling, and translation in time-series databases. In Proceedings of the 21th International Conference on Very Large Data Bases, Zurich, Switzerland, 11–15 September 1995; pp. 490–501.
34. Bingham, S.; Kot, M. Multidimensional trees, range searching, and a correlation dimension algorithm of reduced complexity. *Phys. Lett. A* **1989**, *140*, 327–330. [CrossRef]
35. Bellman, R. *Adaptive Control Processes: A Guided Tour*; Princeton University Press: London, UK, 1961; Volume 4.
36. Verleysen, M.; François, D. The Curse of Dimensionality in Data Mining and Time Series Prediction. In Proceedings of the 8th International Work-Conference on Artificial Neural Networks, Barcelona, Spain, 8–10 June 2005; pp. 758–770.
37. Yianilos, P.N. Data structures and algorithms for nearest neighbor search in general metric spaces. In Proceedings of the 4th annual ACM-SIAM Symposium on Discrete Algorithms, Austin, TX, USA; 1993; pp. 311–321.
38. Bozkaya, T.; Ozsoyoglu, M. Indexing large metric spaces for similarity search queries. *ACM Trans. Database Syst. (TODS)* **1999**, *24*, 361–404. [CrossRef]
39. Uhlmann, J.K. Satisfying general proximity/similarity queries with metric trees. *Inf. Process. Lett.* **1991**, *40*, 175–179. [CrossRef]
40. Nair, A.S.; Sreenadhan, S.P. A coding measure scheme employing electron-ion interaction pseudopotential (EIIP). *Bioinformation* **2006**, *1*, 197. [PubMed]
41. Holden, T.; Subramaniam, R.; Sullivan, R.; Cheung, E.; Schneider, C.; Tremberger, G.; Flamholz, A.; Lieberman, D.H.; Cheung, T.D. ATCG nucleotide fluctuation of Deinococcus radiodurans radiation genes. In Proceedings of the Instruments, Methods, and Missions for Astrobiology X, San Diego, CA, USA, 1 October 2007.
42. Voss, R.F. Evolution of long-range fractal correlations and 1/f noise in DNA base sequences. *Phys. Rev. Lett.* **1992**, *68*, 3805. [CrossRef] [PubMed]
43. Faloutsos, C.; Ranganathan, M.; Manolopoulos, Y. Fast subsequence matching in time-series databases. In Proceedings of the 1994 ACM SIGMOD International Conference on Management of data, Minneapolis, MN, USA, 24–27 May 1994.

44. Mitsa, T. *Temporal Data Mining*; CRC Press: New York, NY, USA, 2010.
45. Mörchen, F. *Time Series Feature Extraction for Data Mining Using DWT and DFT*; Technical Report 3; Departement of Mathematics and Computer Science Philipps-University Marburg: Marburg, Germany, 2003; pp. 735–739.
46. Jensen, A.; la Cour-Harbo, A. *Ripples in Mathematics: The Discrete Wavelet Transform*; Springer: Berlin, Germany, 2001.
47. Wu, Y.-L.; Agrawal, D.; El Abbadi, A. A comparison of DFT and DWT based similarity search in time-series databases. In Proceedings of the 9th International Conference on Information and Knowledge Management, Washington, DC, USA, 6–11 November 2000; pp. 488–495.
48. Caboche, S.; Audebert, C.; Lemoine, Y.; Hot, D. Comparison of mapping algorithms used in high-throughput sequencing: Application to Ion Torrent data. *BMC Genom.* **2014**, *15*, 264. [CrossRef] [PubMed]
49. Cotten, M.; Petrova, V.; Phan, M.V.; Rabaa, M.A.; Watson, S.J.; Ong, S.H.; Kellam, P.; Baker, S. Deep sequencing of norovirus genomes defines evolutionary patterns in an urban tropical setting. *J. Virol.* **2014**, *88*, 11056–11069. [CrossRef] [PubMed]
50. Phan, M.V.; Anh, P.H.; Cuong, N.V.; Munnink, B.B.O.; van der Hoek, L.; My, P.T.; Tri, T.N.; Bryant, J.E.; Baker, S.; Thwaites, G. Unbiased whole-genome deep sequencing of human and porcine stool samples reveals circulation of multiple groups of rotaviruses and a putative zoonotic infection. *Virus Evol.* **2016**, *2*. [CrossRef]
51. Kiyuka, P.K.; Agoti, C.N.; Munywoki, P.K.; Njeru, R.; Bett, A.; Otieno, J.R.; Otieno, G.P.; Kamau, E.; Clark, T.G.; van der Hoek, L. Human Coronavirus NL63 Molecular Epidemiology and Evolutionary Patterns in Rural Coastal Kenya. *J. Infect. Dis.* **2018**, *217*, 1728–1739. [CrossRef] [PubMed]
52. Arias, A.; Watson, S.J.; Asogun, D.; Tobin, E.A.; Lu, J.; Phan, M.V.; Jah, U.; Wadoum, R.E.G.; Meredith, L.; Thorne, L. Rapid outbreak sequencing of Ebola virus in Sierra Leone identifies transmission chains linked to sporadic cases. *Virus Evol.* **2016**, *2*. [CrossRef] [PubMed]
53. Agoti, C.N.; Otieno, J.R.; Munywoki, P.K.; Mwihuri, A.G.; Cane, P.A.; Nokes, D.J.; Kellam, P.; Cotten, M. Local evolutionary patterns of human respiratory syncytial virus derived from whole-genome sequencing. *J. Virol.* **2015**, *89*, 3444–3454. [CrossRef] [PubMed]
54. Altschul, S.F.; Gish, W.; Miller, W.; Myers, E.W.; Lipman, D.J. Basic local alignment search tool. *J. Mol. Biol.* **1990**, *215*, 403–410. [CrossRef]
55. Menzel, P.; Ng, K.L.; Krogh, A. Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nat. Commun.* **2016**, *7*, 11257. [CrossRef] [PubMed]
56. Langmead, B.; Salzberg, S.L. Fast gapped-read alignment with Bowtie 2. *Nat. Methods* **2012**, *9*, 357. [CrossRef] [PubMed]
57. Sović, I.; Šikić, M.; Wilm, A.; Fenlon, S.N.; Chen, S.; Nagarajan, N. Fast and sensitive mapping of nanopore sequencing reads with GraphMap. *Nat. Commun.* **2016**, *7*, 11307. [CrossRef] [PubMed]
58. Otto, C.; Stadler, P.F.; Hoffmann, S. Lacking alignments? The next-generation sequencing mapper segemehl revisited. *Bioinform.* **2014**, *30*, 1837–1843. [CrossRef] [PubMed]
59. Tapinos, A.; Robertson, D.L. De novo assembly of nucleotide sequences in a compressed feature space. In Proceedings of the 2017 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Manchester, UK, 23–25 August 2017; pp. 1–7.
60. Li, D.; Liu, C.-M.; Luo, R.; Sadakane, K.; Lam, T.-W. MEGAHIT: An ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* **2015**, *31*, 1674–1676. [CrossRef]
61. Anton, B.; Sergey, N.; Dmitry, A.; Alexey, A.; Mikhail, D.; Alexander, S.; Valery, M.; Sergey, I.; Son, P.; Andrey, D. SPAdes: A new genome assembly algorithm and its applications to single-cell sequencing. *J. Comput. Biol.* **2012**, *19*, 455.
62. Tapinos, A.; Mendes, P. A method for comparing multivariate time series with different dimensions. *PloS ONE* **2013**, *8*, e54201. [CrossRef]
63. Sheybani, E.O. An Algorithm for Real-Time Blind Image Quality Comparison and Assessment. *Int. J. Electr. Comput. Eng. (IJECE)* **2011**, *2*, 120–129. [CrossRef]
64. Hendriks, R.C.; Gerkmann, T.; Jensen, J. DFT-domain based single-microphone noise reduction for speech enhancement: A survey of the state of the art. In *Synthesis Lectures on Speech and Audio Processing*; Morgan & Claypool: San Rafael, CA, USA, 2013; pp. 1–80.
65. Kouchaki, S.; Tapinos, A.; Robertson, D.L. A signal processing method for alignment-free metagenomic binning: Multi-resolution genomic binary patterns. *Sci. Rep.* **2019**, *9*, 2159. [CrossRef]

66. Shi, H.; Schmidt, B.; Liu, W.; Müller-Wittig, W. A Parallel Algorithm for Error Correction in High-Throughput Short-Read Data on CUDA-Enabled Graphics Hardware. *J. Comput. Biol.* **2010**, *17*, 603–615. [CrossRef]
67. Zhang, Q.; Pell, J.; Canino-Koning, R.; Howe, A.C.; Brown, C.T. These Are Not the K-mers You Are Looking For: Efficient Online K-mer Counting Using a Probabilistic Data Structure. *PLoS ONE* **2014**, *9*, e101271. [CrossRef]
68. Salikhov, K.; Sacomoto, G.; Kucherov, G. Using cascading Bloom filters to improve the memory usage for de Brujin graphs. *Algorithms Mol. Biol.* **2014**, *9*, 364–376. [CrossRef]
69. Berlin, K.; Koren, S.; Chin, C.-S.; Drake, J.P.; Landolin, J.M.; Phillippy, A.M. Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. *Nat. Biotechnol.* **2015**, *33*, 623–630. [CrossRef]
70. Laver, T.; Harrison, J.; O'neill, P.; Moore, K.; Farbos, A.; Paszkiewicz, K.; Studholme, D.J. Assessing the performance of the oxford nanopore technologies minion. *Biomol. Detect. Quantif.* **2015**, *3*, 1–8. [CrossRef]
71. Fu, S.; Wang, A.; Au, K.F. A comparative evaluation of hybrid error correction methods for error-prone long reads. *Genome Biol.* **2019**, *20*, 26. [CrossRef]
72. Watson, M.; Warr, A. Errors in long-read assemblies can critically affect protein prediction. *Nature Biotechnol.* **2019**, *37*, 124. [CrossRef]
73. Radovanović, M.; Nanopoulos, A.; Ivanović, M. Time-series classification in many intrinsic dimensions. In Proceedings of the 2010 SIAM International Conference on Data Mining, Columbus, OH, USA, 29 April–1 May 2010; pp. 677–688.