

Accelerating Dynamic Time Warping Algorithm: A Review

Denuke Dissanayake

Department of Computer Engineering
University of Peradeniya
Peradeniya, Sri Lanka
e16089@eng.pdn.ac.lk

Nipun Dewanarayane

Department of Computer Engineering
University of Peradeniya
Peradeniya, Sri Lanka
e16360@eng.pdn.ac.lk

Maneesha Randeniya

Department of Computer Engineering
University of Peradeniya
Peradeniya, Sri Lanka
e16313@eng.pdn.ac.lk

Hasindu Gamaarachchi

Garvan Institute of Medical Research
Sydney, Australia
hasindu@garvan.org.au

Roshan Ragel

Department of Computer Engineering
University of Peradeniya
Peradeniya, Sri Lanka
roshanr@eng.pdn.ac.lk

Abstract—Dynamic Time Warping is a widely used algorithm for finding the optimal alignment of two time series. It was introduced in the 60s and popular in the 70s for speech recognition problems. DTW can be applied to any data converted into a linear sequence, like video, audio and graphic data. However, the DTW has a time complexity of $O(n^2)$. This complexity limits the potential of the DTW algorithm. DTW consumes more computational power and time when the time series becomes ultra-long. Over the last decade, the time series research community seems to have concluded that DTW can accelerate more and achieve accurate implementations with less time and space complexity. The algorithm's simplicity and dynamic programming base lead to more accelerated versions and improvements. Some DTW versions are specifically modified for special applications, resulting in more advantages. DTW is still not fully appreciated in the community and has the potential to improve, making it even stronger and more efficient.

Index Terms—Accelerate Dynamic Time Warping, Nanopore Sequencing

I. INTRODUCTION

An algorithm is a set of instructions to perform a specific task or solve a problem. They are mainly used in mathematics and computer science. In computer science, algorithms run a specific task or programme. An algorithm should be suitable when choosing an algorithm to perform a task or solve a problem. There are characteristics to be considered when choosing an algorithm. Time and space complexity, accuracy, and efficiency are some of them.

Dynamic Time Warping (DTW) is one of the most important algorithms for comparing alignments of two time series. One of the advantages factors of the DTW algorithm is that it can measure two time series with different lengths and time shifts. It has many use cases, such as speech recognition, pattern recognition, signature recognition, shape matching etc. There are various proposed software and hardware implementations to optimise the DTW algorithm. We will discuss it more in this review. Other than that, many variations of the DTW algorithm are used in many applications.

One application that uses the DTW algorithm is Nanopore Selective Sequencing, which allows selecting a specific sequence section from a very large sequence (RNA or DNA). Nanopore Technologies introduced the MinION sequencer in 2014. With that, Nanopore Selective Sequencing comes into the picture. Since then Nanopore Selective Sequencing method has been rapidly becoming popular worldwide because of its unique features. We will discuss them in the latter part of the review paper.

The rest of the paper is organised as follows. Section II introduces the DTW algorithm and how two signals are compared using the DTW algorithm. Section III will review some variations of DTW algorithms, how the DTW algorithm is modified related to different domains, and next in section IV, how the DTW algorithm can be customised to improve its performance. We will discuss a few customisation methods used to improve performance. Sections V and VI will discuss how the DTW algorithm can be optimised using software and hardware. In the hardware optimisation section, we mainly focus on GPUs and FPGAs. Finally, in section VII, we will discuss applications of the DTW algorithm. In that section, we review some main applications of DTW and prioritise Nanopore Sequencing; discuss how Nanopore Sequencing is related to DTW and how Selective Sequencing can be done using the DTW algorithm. In the discussion, we show some gaps in the literature. We suggest methods to improve the efficiency of Nanopore Selective Sequencing using the DTW algorithm with the GPUs-based acceleration approach.

II. DYNAMIC TIME WARPING ALGORITHM

Dynamic Time Warping (DTW) [6] [7] is a mathematical algorithm introduced in the 1960s that analyses the similarities and alignments of two time-series. It has a computational complexity of $O(n^2)$. It became a useful algorithm because sequences could be analysed even with shifts and distortions in time-series. It was used in the early stages to identify speech patterns and recognition purposes. But later, it was used

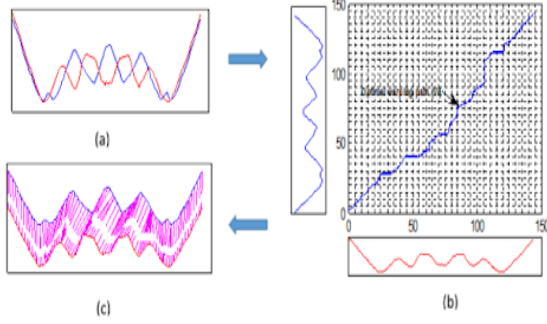


Fig. 1: How the DTW algorithm works. (a) Two time series from the Plane time-series data-set, (b) The optimal warping path and (c) The alignment between these two-time series [42]

for many more purposes such as letter, signature and gesture recognition, handwriting identification, computer vision, computer surveillance, data mining pattern recognition, sequence identification etc.

At First DTW algorithm builds the distance matrix $C \in \mathbb{R}^{N \times M}$ representing all pairwise distances between two sequences. This distance matrix is called the local cost matrix for aligning two sequences. Once the local cost matrix is built, the algorithm finds the alignment path that runs through the cost matrix's low-cost areas. This path built by DTW is a sequence of points [6].

Let's take two time series $S = s_1, s_2, s_3, \dots, s_n$ and $C = c_1, c_2, c_3, \dots, c_n$. Concerning these two time series, DTW distance can be denoted as $D(S, C) = d(n, n)$.

$$D(S, C) = d(n, n)$$

$$d(i, j) = |c_i - q_j| + \min\{d(i-1, j), d(i, j-1), d(i-1, j-1)\}$$

$$d(0, 0) = 0; d(i, 0) = d(0, j) = \infty; i = 1, 2, \dots, n; j = 1, 2, \dots, n$$

The n -by- n matrix is called as cost matrix. The value of each cell is calculated using the values from previously computed three values. The warping path can be obtained by tracing back through calculated values. It gives the optimal alignment between S and C series. The time and space complexity of the algorithm is $O(n^2)$. By deleting the trace back values, space complexity can be reduced to $O(n)$

Many proposed optimisation methods exist for the DTW algorithm, such as step function, weighting and global path constraints [6]. These will add new modifications to the original DTW algorithm. DTW algorithm is successful where time complexity is not an issue. After DTW was used in many fields and used for comparing ultra-long sequences, researchers studied the native DTW. They modified it to new algorithms, reducing time or space complexity. Examples are SDTW (segmented DTW), DDTW (Derivative DTW), FDTW (Fast DTW), Sparse DTW, QDTW (Qualitative DTW) [14]. In the next section, we will discuss more details about variations of the classical DTW algorithm.

III. VARIATIONS OF THE DTW ALGORITHM

This section provides a overview of the most widely used DTW modifications.

In 1957, Richard Bellman discovered the concept of dynamic programming, which can massively improve the efficiency of algorithms. DTW is primarily based on this concept and measures the similarity between two data sequences. In 1978 Sakoe and Chiba used the concept of Dynamic Time Warping for speech recognition. Starting from that, DTW evolved a lot and optimised in various ways [14].

- Double-stage DTW (DSDTW) was presented in 2010 by Adwan and Arof. This version is mostly used in face detection. It was observed that the efficient method in terms of time complexity and accuracy. Two-dimensional images will be converted to one-dimensional vectors. DTW is used to compare two vectors of the two different images. DSDTW reduce the time complexity from quadratic to linear [26].
- Fast search DTW (FDTW) was proposed in 2005, based on the lower bound distance measure also used in DTW. This approach is widely used in sequence matching in DNA and other applications because faster than the existing DTW algorithm [27].
- Subsequence DTW (sDTW) is one of the important improvements of the DTW algorithm used in sequencing. It is very useful in sub-sequencing; when searching for alignments in a small portion of a large sequence. It is widely used in nanopore selective sequencing implementations (ONT MinION sequencer) [19] [20]. Müller first introduced the subsequence DTW algorithm to modify the classical DTW. This algorithm is specifically made to compare input subsequence (X) with index full sequence (Y). This will identify the optimum matching subsequence from the index sequence. Mostly used in query-by-example applications. First, calculate the accumulated cost matrix, where the size M of the input sequence and the size of the N index sequence ($M \ll N$).

$$D(m, n); 1 \leq m \leq M; 1 \leq n \leq N$$

Once completely calculating matrix D , find the optimal subsequence of index where start point is a^* and end point is b^*

$$(a^*, b^*) := \underset{(a, b): 1 \leq a \leq b \leq M}{\operatorname{argmin}} (DTW(X, Y(a : b)))$$

To find optimum subsequence endpoint,

$$b^* := \underset{b \in [1: M]}{\operatorname{argmin}} D(N, b)$$

Like this starting point, a^* can be found using $D(1, a)$ instead of $D(N, b)$. Therefore optimal matching subsequence can be found as $Y(a^*, b^*)$ [19] [20].

- Spare DTW was presented by Al-Naymat et al. in 2009 to reduce the space complexity of the DTW algorithm, With this method, it will reduce from quadratic to linear [28].
- Qualitative DTW(QDTW) is another variation of this algorithm to transform numerical time series into quanti-

tative ones. These quantitative time series are shorter than numerical time series. Because of that, when applying the DTW algorithm to qualitative time series will give higher speed and high accuracy when compared to numerical time series [29].

- Flexible DTW (FDTW) was proposed for the one-to-one mapping of long sequences. When compared to the DTW algorithm, it will give less error rate. It will avoid matching points too far from each other in the time series [30].
- Space Efficient DTW was proposed to improve speech recognition performance by Darabkh, Khalid A. et al.in 2009 [31].
- Five Stage DTW was proposed to reduce the time complexity from quadratic to linear in isolated word recognition [32].
- Iterative deepening DTW (IDDTW) was proposed to measure the lower boundings of a time series. But this approach is based on a probabilistic model.
- In 2007 Salvador, S. Chan, P. used the method ‘Divide at Impera’ to improve the DTW algorithm. This method divides distance matrices into 2, 4, 8, 16, ... etc. This approach always gives an optimal result for the distance between two time series [33].
- In 2009 Mizutani Eiji proposed a new method to overcome the multi-stage optimisation problem; the state is transformed from one step to another [34].
- Weighted DTW (WDTW) Jeong suggests weighted DTW (WDTW), a penalty-based DTW [38]. This variation’s main characteristic is that time-series points are not all handled equally. A test point and a reference point’s phase differences are used to provide weights to the points. The local distance between the i th and j th locations of Q and C , respectively, is determined using the WDTW technique as follows:

$$d(q_i, c_j) = W|i - j|(q_i - c_j)$$

In this case, the weight penalty value between the two points, q_i and c_j , is represented as $w|i - j|$.

- Derivative DTW (DD-DTW) - Grecki and uczak developed another variant of DTW called DD-DTW.[39] It combines a value-based DTW with a derivative-based DTW using parameters. This approach separately computed the DTW on raw time-series and the DTW on derivative time-series distances. Each distance is then given a weight value. The weighted distances’ sum is then provided.

$$DD - DTW(Q, C) = a * DTW(Q, C) + b * DTW(Q0, C0)$$

Where, (Q, C) are the original time-series, (Q, C) are their derivative time-series respectively, and $0 \leq (a, b) \leq 1$.

- Shape DTW (shapeDTW) - The shapeDTW approach is presented by Zhao and Itti to improve the alignment quality of the classical DTW [40]. This variation’s concept generates rich information about the local shape by

examining nearby points.

- Locally slope DTW (LSDTW) - The methodology is based on local slope features and is known as Locally Slope DTW (LSDTW) [41]. By using the local neighbourhood information surrounding each point, LSDTW tries to match nearby similar shapes. There are three main steps in LSDTW. To address the noise issue, a filtering technique is applied initially. Second, nearby points derive local slope features from each point. Then, the local slope characteristics are symbolised. The resulting representations are then subjected to DTW to determine the best alignment.

These variations are introduced in different applications to achieve specific improvements in the DTW algorithm for each domain. The main goal of these improvements is to reduce the time and space complexity of the DTW algorithm.

IV. CUSTOMISATIONS OF DTW ALGORITHM

DTW algorithm is used to find the optimal alignments of two time series. But it is complex and slow due to its quadratic behaviour. Because of that, researchers have found some approaches to reduce its complexity.

A. Lower Bounding functions

Lower bounding functions improve the efficiency of DTW because they can quickly prune sequences that could not be the best match. Those functions have two key properties: incur no false dismissals, and the tighter they are, the more they prune. More pruning will cause less accuracy in DTW. LB_Keogh is well known lower bound function that rescales the two matching sequences to the same length. LB_PPAA is another approach that requires chopping all sequences to map with reference sequences that create down-sampled sequences along their timeline. These are compared with the down-sampled reference sequence. LB_Keogh and LB_PAA look the same, but in LB_Keogh, sequences are chopped to the minimum length of sequence going to compare, but in LB_PAA, chopped for custom N value, which is smaller than the previous one. So calculating LB_PAA is more efficient than LB_Keogh. Both methods need to be computed for each comparing sequences from the database to find the most similar to the reference sequence [18].

B. Indexing Time Series

Researchers propose indexing techniques to compare a small subset of relevant sequences with the reference sequences. That prunes the search space even more. LB_PAA convert sequences into N -dimensional space which helps to index them. For indexing, R-Tree is a proposed method. The index contains minimum bounding rectangles (MBR) in which similar sequences are grouped. The algorithm can find the sequences from the database close to the reference sequence using the proximity of their corresponding MBR from the index [18].

C. Approximate Searches

Approximate searches achieved a great high performance in response time with some potential false dismissals. Approximations use to assess the similarity between time series. But the drawback of this method is that there is a tradeoff between accuracy and the response time [18].

D. Constrained DTW paths

After calculating the cost matrix, DTW will create a path going through the low-cost points. Some applications are restricting the admissible paths around the diagonal. This will reduce the complexity of the path calculation, and in practice, this will increase the accuracy also. The most used constraints are the Sakoe-Chiba band Itakura parallelogram. In Sakoe-Chiba, admissible paths are restricted inside a band of fixed width. Itakura uses a parallelogram instead of a fixed-width band [15] [18].

V. OPTIMISATIONS AND IMPROVEMENTS OF DTW ALGORITHM - SOFTWARE

This section will discuss software-based improvements to the classical DTW algorithm to improve efficiency and accuracy.

A. Subsequence Pattern Mining

The novel algorithm can identify the two time series subsequences. The intuitive algorithms for subsequence mining apply the DTW algorithm to all subsequences in two time series. The proposed method assumes that if the distance between two data points is large, it is less likely to lead to a similar pattern. Start the algorithm, and first determine the low-cost values of the inputs (time series). It uses a threshold value; if the cost is less than the threshold, consider a possible match and start/continue a DTW calculation.

$$(i, j) \quad (1 < i < N \text{ and } 1 < j < M)$$

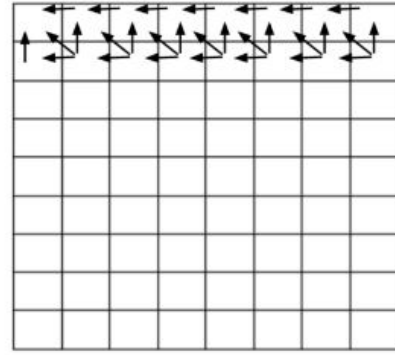
$$(X, Y \text{ are time series with size of } N, M)$$

$$\frac{DTW_{Distance}}{DTW_{Length}} < threshold ,$$

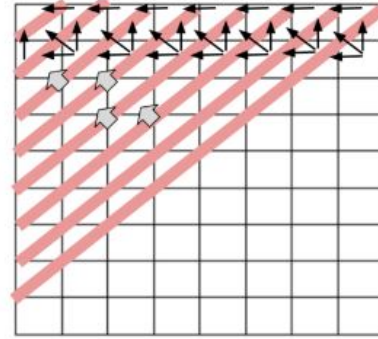
$$DTW_{distance} = \sum_{i,j \in \text{warping path}} \text{cost}(x_i, y_j)$$

$$DTW_{length} = \text{length of the warping path}$$

There is a common issue in most of the DTW-based algorithms, that is over warping in the warping path. To avoid that, window constraint is imposed when computing the warping path. Also, this window constrains the maximum size of the subsequence which can match. There is an offset threshold to control the size of the warping window. The computation of the DTW will stop when the point reaches the warping window boundary, and the warping path will not increase the length of the warping window. In some cases, matching subsequences can be too small and can not consider a valid match. In that case, the algorithm uses another threshold to filter too small subsequences. In that case, the subsequence should be bigger than the threshold to be considered a valid match [15].



(a) Computation dependencies



(b) Diagonal elements of the cost matrix

Fig. 2: [15]

B. Parallelized DTW

Naive Dynamic Time Warping algorithm is based on a dynamic programme technique, which stores the previously computed results in a matrix. Most of the matrix calculations can be parallelised and do calculations more efficiently. Also, DTW does not require complex mathematical models, so it is easy to divide the process into simple calculations and increase efficiency. However, the computation of each iteration in the matrix depends on the result from the last step, which will limit the parallelisation ability during the computation. There are several strategies to improve the degree of parallel ability in DTW [15] [16].

The cost matrix of the DTW in each cell represents the DTW distance between two data points. The computation of each cell depends on the results of the left, top, and top-left cells (Fig. 2a). For cell (i,j) in the matrix D , value of $D[i, j]$ depends on the values in $D[i, j-1]$, $D[i-1, j]$, $D[i-1, j-1]$. In that case, for any cell (i,j) in the matrix, all cells between $(0,0)$ and (i,j) need to be computed before calculating the value of the position (i,j) . The DTW algorithm performs this computation in row or column order. The cells in the same row or column are computed in sequence.

On the other hand, there is no data dependency between the diagonal cells. Therefore, diagonal sets of cells can be computed in parallel. The data dependency exists between two neighbouring diagonals. Therefore, we can only compute the

data within one diagonal set at a time (Fig. 2b). The flow of the method is calculation starts with the first cell and then updates the cells in the closest diagonal line set. Since these cells inside the diagonal only depend on the first cell, they can be computed in parallel. After finishing the diagonal set, move to the closest diagonal set sequentially. Follow through until the matrix gets filled. This strategy guarantees data dependency, and reasonable parallelism can be achieved.

C. Continuous Wavelet DTW

The original DTW algorithm has a complexity of $O(n^2)$. This will affect significantly when it comes to the very long sequences, not only that the frequency of measuring current is higher than the passing of the DNA sequences, which creates a magnitude difference in the expected signal sequence and the sampling rate of the raw signal.

This research proposed an alternative method for the traditional DTW, called Continuous Wavelet Dynamic Time Warping (cwDTW), based on Continuous Wavelet Transform (CWT) [13]. The principle behind this approach is to take a series of signals with different resolution levels using the CTW. With CWT's help in considering their sampling speed, the transformed coarsening signal sequences from the two input sequences with different sampling rates would have comparable lengths and similar shapes at each resolution level. Also, this method can reduce the details of highly representative coarsening signals and help DTW compare two sequences efficiently. According to the calculations, cwDTW bounded within $O(N \log N)$ for the extreme cases [12].

VI. ACCELERATION OF DTW ALGORITHM - HARDWARE USING GPUS AND FPGAS

In this section, we mainly discuss two methods of DTW accelerations for Subsequence Similarity Search. Subsequence Similarity Search matches a specific region of time series from a long time series.

The first method is using Graphics Processing Units (GPUs). GPUs are processors designed to process and accelerate graphics renderings. But it can be used to do general purpose calculations. When compare to CPUs, GPUs have more cores than CPUs but cores are less powerful. These large number of core GPUs can run the calculations in parallel and speed up the execution process. NVIDIA CUDA can be considered one of the most popular GPU programming tool kits. The proposed method consists of three steps in implementing the DTW in GPUs. 1. Copy the data into the GPU memory. In this step, the raw signal or time series is copied from CPU memory to the GPU memory. If the signal exceeds the GPU memory, it will process as small batches. 2. Execute the code in the GPU kernel. In this step, mainly two things happen. First, find the mean and the variance. Second, find the normalised DTW distance to queries. After the mean and variance are calculated, each thread calculates the DTW distance between the query and the normalised subsequence. 3. Copy back the output to the CPU memory. This will accelerate the DTW algorithm compared to the CPU execution.

The second method is optimising using Field-Programmable Gate Arrays (FPGA). The proposed FPGA design consists of two parts; Normalizer and Wrapper. Those input data are first normalised, and then the DTW algorithm calculations are performed. Input data is streamed into a FIFO buffer, and a sample is generated as output in every clock cycle. This output is fed onto the Normalizer, and Normalizer is waiting for the first window to be received. After the first output is generated at every clock cycle, a new output is generated. This output is stored in an intermediate buffer between Normalizer and the Wrapper. Wrappers are implemented as a systolic array. Data processing units are connected in a matrix fashion inside the wrapper. The wrapper module generates the DTW distance between the normalised window and the query time series every n (window length) clock cycle [17].

VII. APPLICATIONS OF DTW ALGORITHM

In this section, we discuss applications of the DTW algorithm in different domains. From those applications, we will give a higher focus to Nanopore Selective Sequencing and how DTW is applied with it.

A. Speech and Voice Command Recognition

The Dynamic Time Warping algorithm was widely used in speech recognition in the 70s. Speech recognition is identifying the words humans speak by converting the sound into an electronic signal captured by an audio device. Then translate them into computer-understandable data. This is done by converting the voice signal into numbers to identify each word separately.

DTW algorithm is applied to voice and command identification. DTW will compare a sample set of unknown signals with template signals. In this example, each pattern (voice signal) has some specific waveform, input waveform will check with these patterns and get how many similarities these patterns gain with existing voice signals. Then it can categorise the input signal into a specific voice signal which has the most similarities with the category. In this application, there are some drawbacks due to the DTW algorithm. Calculating DTW for more detailed audio signals requires more computational power and decreases the speed. That also causes the unavailability of real-time processing. In this method, vocabulary has to be restricted, which means the number of specific voice signals to be matched will also reduce. In that case, researchers mainly used GPU parallelisation to overcome the previously mentioned drawbacks [21] [22].

B. Data mining Applications

In the field of data mining, Clustering and Classification play a major part. Both these clustering and classification algorithms are directly dependent on the quality of the distance measuring process. An accurate distance measuring algorithm can improve the overall accuracy. Most recently research community has had more interest in DTW as the best distance measuring algorithm [24]. The community believes that DTW still has more potential to improve its performance and

accuracy, and there are existing optimisations that are great additions to the classical DTW. In the data mining domain, the most important parameter of the DTW is allowable warping width. An increase in the warping window width will result in the elasticity of the algorithm, which allows the algorithm to find similarities in a wide range. The default time series distance measurement method is the Euclidean distance method, a special case of DTW where warping width is zero. In Euclidean distance, only consider the direct point-to-point distance. Here DTW with more warping path decreases the error rate, for example, data sets. In some examples, when warping width increases continuously, the error rate decreases and the test and train rate accuracy will increase. In some example data sets, the error rate will increase after some value of the warping width [23]. So larger values of warping width can make things better or worse. Increasing the warping width unnecessarily will cause a huge computational burden for no effect. Practical results in the data mining indicate blindly using Euclidean distance will lead to poor results on some data sets and can improve them massively by use of DTW with correct warping width. There is a proposed way to learn the value of the best warping width for a sample. Use cross-validation on the labelled training data to examine the error rate for all values of warping width, then choose the one that minimises the predicted error rate [25].

C. Signature verification using DTW

Signatures are special classes of handwriting in which words and letters may not be identified clearly, and these characters incorporate complex geometrical patterns. In signature verification, a person has to write the signature relatively same over time; if two signatures have more mismatches, it will be identified as a copied or incorrect signature. After pre-processing the images of the signatures, 1-D features can be extracted. The typical 1-D features extracted for signature images include vertical projection, horizontal projection, top contour, envelope width and contour ratio. Among that, vertical projection is the most used one. The vertical projection is the sum of pixels along the columns of a binary image. In DTW, it will find the optimal match between two feature vectors extracted from two signatures. Non-linear matching between two feature vectors can be achieved and create a cost matrix. After calculating the warping path and its warping cost, dissimilarities between the two signatures can be measured. We can find the minimum allowable dissimilarity for two signatures using an example dataset. An offline handwritten signature was proposed that uses the 1-D feature with DTW. Also, some modifications were made in the original scenario, and the modified DTW algorithm performed significantly better than the basic DTW [35].

D. Word recognition

Since the introduction of the DTW algorithm, it has been popular for speech recognition applications. Later it was discovered that the DTW algorithm could be used in applications in word recognition. There are many applications associated

with word recognition; isolated, connected, and discrete word recognition are some of those.

The proposed method for connected word recognition is an improvement of a 'stack decoding algorithm' [36] originally proposed for continuous speech recognition applications. The major difference between those two approaches is in the earlier method, a probabilistic finite state model is used, while in this new approach, a signal processing formulation. In order to recognise continuous words, it is necessary to have a reference pattern for each word (isolated words) in the vocabulary. Then the connected words are identified by connecting each reference pattern (connects isolated words). The role of the DTW algorithm in word recognition is to provide the time alignments between speech words' signals and the reference signals and select the matching reference to the input signals.

To identify the isolated words set of improvements has been made to Sakoe and Chiba's Dynamic Time Warping algorithm [37]. With those modifications, the new algorithm gives better performance in recognition accuracy as well as computational complexity. The main difference between the two approaches is the adjustment window. The old algorithm has an adjustment window restricting the warping function, and the equation gives the adjustment window $|i-j| \leq r$ (is a positive integer) where r is the window length. In the new proposed algorithm adjustment window is given by $|i - (j/s)| \leq r$, where s is the slope of the line connecting point (0,0) and (I, J) (ending point). Though the change is small, the results will show it will significantly affect the algorithm's results.

E. Nanopore Sequencing

The history of Nanopore Sequencing goes back to the 1980s. From that period to today, many improvements have been made to Nanopore Sequencing. The concept involves a nanopore (a nano-sized pore/ protein pore), DNA or RNA molecules and driving motor protein. A constant voltage is applied to the nanopore embedded in a polymer membrane. Then, negatively charged DNA or RNA molecules are driven through the nanopore using the motor protein from the negative side ('cis') to the positive side ('trans') side of the nanopore. The speed can be controlled using the motor protein. The motor protein drives the molecule through the nanopore step by step. With molecules driving through the nanopore, it changes the current corresponding to the nucleic acid in the DNA or RNA molecule. These changes in the region of the nucleonic sequence which passes through can be monitored and decoded using mathematical algorithms [1] [3].

Throughout history, there have been many advancements in Nanopore Sequencing. One of the breakthroughs was the introduction of the MinION sequencer in 2014 by Oxford Nanopore Technologies (ONT). Compared to the Pacific Biosciences (PacBio), ONT's MinION has more advantages, such as its compact smaller size and portability. Earlier it was much larger machines and sequencing centres with PacBio's technologies. But in contrast, MinION is a USB-powered small device around four inches long. It was less expensive

than maintaining servers; anyone could buy a MinION and do their sequencing. It only needed a laptop to connect the MinION sequencer, and the resulting data could be easily stored in an external hard drive for further use. Earlier samples needed to be brought to the sequencing centres to do the experiments, but the MinION toolkit could be brought to the laboratory or the fields. This is a considerable advantage over PacBio technology. Another significant benefit of MinION is it gives real-time information [2] [3].

1) *Nanopore Selective Sequencing*: Selective Sequencing is a special feature of ONT, enabling users to select a specific sequence from a DNA or RNA molecule. ‘Read Until’ is an Application Programming Interface (API) introduced by the ONT to achieve selective sequencing. It reverses the voltage provided across the nanopore and rejects a specific sequence as the user needs, enabling selectively selecting a specific section of the molecule. Selective Sequencing enables real-time data capturing capabilities. Because of that, it is no need to wait until the whole DNA or RNA molecule is complete sequenced [2] [4].

ONT sequencers support selective sequencing. Researchers have implemented a new streaming method of converting raw signals to real-time selective sequencing. As the common method, base callers convert raw signals to nucleotide sequences. But this approach involves computationally high demand calculations, which take many times to sequence. To overcome these issues, it is necessary to use high-performance computers. Otherwise, it may take days in CPUs and hours in GPUs (~20 Gbp data). This throughput is not sufficient when quick results are needed. This makes the portability of sequencers invalid because samples need to be taken to the laboratories to get fast results that do not allow check results in the field itself. This problem is overcome to some extent by introducing selective sequencing and using the DTW algorithm to selectively select the gnomes. This method also works for the gnome size of kilo base pair long. This happens because base callers are not optimised well to perform the sequencing on small chunks.

2) *Nanopore Sequence analysing methods*: Nanopore sequencing able to generate extremely long reads, like from 10 to 100 kbp, without any PCR amplification. There are k number of consecutive nucleotides in a pore at single time point (referred as a k-mer) and the electrical signal is measured for this k number of nucleotides in a pore at each time. In this process, raw electrical current will be generated [12]. There are mainly two methods to compare a DNA sequence with a reference sequence. The most common is translating the reads into a nucleotide sequence by base calling. This base calling method is for an electrical signal value to the corresponding nucleotide sequence. Following this, the signal will generate a nucleotide sequence [11]. Initially using hidden Markov models, ONT has created a number of base-callers for nanopore sequence data that are accessible via the Metrichor cloud service [5]. These were swapped out for neural network models that ran on CPUs and GPUs. ONT offers a variety of computational platforms with integrated GPUs for real-time

base-calling (minIT, Mk1C, GridION and PromethION). Real-time base-calling is made possible by these gadgets so that flow cells can continue to generate data. Recently, these base-callers acquired a server-client setup, enabling them to send the raw signal to the server and receive back a nucleotide sequence. [4] [5].

Another method for nanopore sequence analysis is the end-to-end mapping between raw electrical signal sequence and reference sequence signal. The most widely-used and powerful solution to find an optimal mapping for signal-to-signal alignment is Dynamic Time Warping (DTW). However, the original DTW has an $O(N_1 N_2)$ time and memory complexity, where N_1 and N_2 are the lengths of the two sequences to be mapped [12]. Also, DTW is guaranteed to find the optimal alignment of two series of time-ordered data, and it has previously been used to analyse sequence data [5].

3) *Nanopore Selective Sequencing using DTW*: This concept matches the reading sequence directly to a reference sequence in real-time using the DTW algorithm. The algorithm enables the matching of two sequences while MinION manipulates the output sequence. In this approach, the ‘Read Until’ script (a computational tool. Not the ‘Read Until’ concept of ONT) is used to selectively select the specific region from the sequence (The scripts are available to researchers freely under MIT licence: <https://github.com/mattloose/RUscripts>). Environment changes, voltage variations, noise, and interactions with other channels may cause disturbance in matching sequences. To measure, the disturbances are captured using three variables: shift, scale and drift. Z-score transformations can be used to overcome the shift and scale. Drift can be neglected in short sequences. So that using z-score transformations, external disturbances can be omitted [5].

4) *Existing Drawbacks of the Nanopore Sequence analysing using DTW*: Both methods have drawbacks (base calling, DTW). The main disadvantage of base calling is the slow process, but the ONT team rapidly improves their base callers’ efficiency and accuracy [5]. In base calling, the raw signal will translate to read levels. A specific raw signal value will convert to the sequence of nucleotides (k-mers). Such a read-level procedure can miss the unique details connected to an important genomic event, such an epigenetic event. In addition, considerable base-calling errors may be introduced during the conversion of raw current signals to readings. (10% to 25% as reported).

The DTW algorithm has less error because it directly compares the raw signal with the reference signal. The applicability of DTW in many situations involving extremely lengthy sequences are strongly constrained by its significantly greater time complexity, which has an ($O(n^2)$) complexity. The raw signal sequence from the nanopore and the translated predicted signal sequence, however, are both extremely long. An order of magnitude mismatch exists between the sampling rates of the raw signal sequence and the translated predicted signal sequence because the frequency of electrical current measurements is typically 7-9 times higher than the passage speed of the DNA sequence. It is also difficult for DTW to resolve

two input sequences with such a significant sampling rate and length disparity. After considering these facts, comparing a raw electrical signal is more efficient, but there are major drawbacks to overcome. DTW is a more suitable algorithm for the situation and has a history of comparing sequences and more optimisations to increase efficiency and accuracy.

Other than the DTW approach, many other approaches are available for Nanopore Sequencing. Such as Nanopore Selective Sequencing using Deep learning [4], UNCALLED method [9] [10], Sigmap Method [8] [9], and Readfish toolkit [11] are some of them.

VIII. DISCUSSION

This paper reviews the Dynamic Time warping algorithm, how it was accelerated using different techniques and its applications. DTW is a powerful tool when we consider time series analysing. The warping window of the DTW algorithm results in more flexibility in finding the similarities between time series, which has also had a good impact on data mining. However, the time complexity of the algorithm becomes a restriction for it. But many improvements were proposed to reduce the algorithm's time complexity, as we discussed in the Continuous Wavelet DTW section. Parallelizing the process is also challenging because most calculations depend on previous results. As a solution, a proposed Diagonal calculation method increases the potential of parallel processing in DTW. DTW was used in speech recognition applications in earlier stages by comparing voice signals. A lot of improvements and methods have been proposed by the research community, like reducing the resolution of the signals, warping constraints, mining subsequence etc. After that, there were more applications in signature verification, face detection, shape matching, motion identifier etc. DTW had more recognition, and it was used for various applications. As a result, more optimisations were suggested like windowing techniques, subsequence DTW, pruning before calculations, fast DTW, etc. Some of the optimisations are specifically designed for specific applications. When DTW is used to analyse advanced and long-time series like DNA sequence analyses, and advanced data mining applications, time complexity becomes a critical restriction. Parallel processing is a good solution for higher complexity calculations, especially matrix calculations. Start implementing DTW in GPU platforms like CUDA and try to parallelise the DTW calculation.

With these findings, we identify a gap still not addressed in Nanopore Sequencing, which can be solved using the DTW algorithm approach using GPUs. Since Selective Sequencing has more advantages than regular sequencing techniques, it is becoming more popular. Not only that, MinION is a device meant to be portable, which enables one to carry it into the fields and do the sequencing. However, base calling methods are computationally expensive and give relatively low throughput. When molecules get bigger, samples must be carried to the laboratories, and larger servers with high computation powers need to be connected to sequencing the samples, violating the sequencer's portability. Especially when using the DTW

algorithm, its $O(n^2)$ complexity only works well with kilo base pair-long genomes. To get maximum throughput and efficiency, DTW should be more optimised for Selective Sequencing. We suggest that the DTW algorithm can be accelerated using Graphical Processing Units(GPUs) to get better results in selective sequencing. Through that make, the DTW algorithm runs in GPU-equipped devices like laptops.

IX. CONCLUSION

Although the Dynamic Time Warping algorithm was initially proposed as a method to find alignments of two time series, now it has become an algorithm using a wide range of domain areas. With this popularity, many improvements and modifications have been made to the algorithm. Not only that many different algorithm versions have been introduced. There are software and hardware accelerations methods proposed in DTW algorithm optimisation. One of the most popular methods of accelerating is GPU-based acceleration using CUDA. With those improvements and modifications, the DTW algorithm is widely used in many application areas, such as data mining and time series database search, surveillance, chemical engineering, music industry, computer vision etc.

REFERENCES

- [1] Wang, Y., Zhao, Y., Bollas, A., Wang, Y., Au, K. F. (2021). Nanopore sequencing technology, bioinformatics and applications. *Nature biotechnology*, 39(11), 1348-1365.
- [2] Leggett, R. M., Clark, M. D. (2017). A world of opportunities with nanopore sequencing. *Journal of experimental botany*, 68(20), 5419-5429.
- [3] Deamer, D., Akeson, M., Branton, D. (2016). Three decades of nanopore sequencing. *Nature biotechnology*, 34(5), 518-524.
- [4] Danilevsky, A., Polsky, A. L., Shomron, N. (2021). Real-time selective sequencing using nanopores and deep learning.
- [5] Loose, M., Malla, S., Stout, M. (2016). Real-time selective sequencing using nanopore technology. *Nature methods*, 13(9), 751-754.
- [6] Senin, P. (2008). Dynamic time warping algorithm review. Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA, 855(1-23), 40.
- [7] M. Müller, M. (2007). Dynamic time warping. *Information retrieval for music and motion*, 69-84.
- [8] Zhang, H., Li, H., Jain, C., Cheng, H., Au, K. F., Li, H., Aluru, S. (2021). Real-time mapping of nanopore raw signals. *Bioinformatics*, 37(Supplement1), i477-i483.
- [9] Ferragina, P., Manzini, G. (2005). Indexing compressed text. *Journal of the ACM (JACM)*, 52(4), 552-581.
- [10] Kovaka, S., Fan, Y., Ni, B., Timp, W., Schatz, M. C. (2021). Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED. *Nature biotechnology*, 39(4), 431-441.
- [11] Payne, A., Holmes, N., Clarke, T., Munro, R., Debebe, B. J., Loose, M. (2021). Readfish enables targeted nanopore sequencing of gigabase-sized genomes. *Nature biotechnology*, 39(4), 442-450.
- [12] Han, R., Li, Y., Gao, X., Wang, S. (2018). An accurate and rapid continuous wavelet dynamic time warping algorithm for end-to-end mapping in ultra-long nanopore sequencing. *Bioinformatics*, 34(17), i722-i731.
- [13] Torrence, C., Compo, G. P. (1998). A practical guide to wavelet analysis. *Bulletin of the American Meteorological society*, 79(1), 61-78.
- [14] Yadav, M., Alam, M. A. (2018). Dynamic time warping (dtw) algorithm in speech: a review. *International Journal of Research in Electronics and Computer Engineering*, 6(1), 524-528.
- [15] Zhu H, Gu Z, Zhao H, Chen K, Li C, He L (2018). Developing a pattern discovery method in time series data and its GPU acceleration, 1(4), 266-283.

- [16] Gustavo Poli, Joao F. Mari, Jose Hiroki Saito, Alexandre L. M. Levada (2007). Voice Command Recognition with Dynamic Time Warping (DTW) using Graphics Processing Units (GPU) with Compute Unified Device Architecture (CUDA), 1550-6533
- [17] Sart, D., Mueen, A., Najjar, W., Keogh, E., Niennattrakul, V. (2010, December). Accelerating dynamic time warping subsequence search with GPUs and FPGAs. In 2010 IEEE International Conference on Data Mining (pp. 1001-1006). IEEE.
- [18] Tavenard, R., Amsaleg, L. (2015). Improving the efficiency of traditional DTW accelerators. *Knowledge and Information Systems*, 42(1), 215-243.
- [19] Anguera, X., Ferrarons, M. (2013, July). Memory efficient subsequence DTW for query-by-example spoken term detection. In 2013 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). IEEE.
- [20] Tsinaslanidis, P. E. (2018). Subsequence dynamic time warping for charting: Bullish and bearish class predictions for NYSE stocks. *Expert Systems with Applications*, 94, 193-204.
- [21] Poli, G., Mari, J. F., Saito, J. H., Levada, A. L. (2007, October). Voice command recognition with dynamic time warping (dtw) using graphics processing units (gpu) with compute unified device architecture (cuda). In 19th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD'07) (pp. 19-25). IEEE.
- [22] Permanasari, Y., Harahap, E. H., Ali, E. P. (2019, November). Speech recognition using dynamic time warping (DTW). In *Journal of Physics: Conference Series* (Vol. 1366, No. 1, p. 012091). IOP Publishing.
- [23] Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., Keogh, E. (2008). Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2), 1542-1552.
- [24] Bagnall, A., Lines, J., Vickers, W., Keogh, E. (2018). The UEA UCR time series classification repository. URL <http://www.timeseriesclassification.com>, 122.
- [25] Dau, H. A., Silva, D. F., Petitjean, F., Forestier, G., Bagnall, A., Mueen, A., Keogh, E. (2018). Optimizing dynamic time warping's window width for time series data mining applications. *Data mining and knowledge discovery*, 32(4), 1074-1120.
- [26] Adwan, S., Arof, H. (2010). A Novel Double Stage Dynamic Time Warping Algorithm for Image Template Matching. In *Proceeding of the 6th IMT-GT Conference on Mathematics, Statistics and its Applications (ICMSA 2010)*, University Tunku Abdul Rahman, Kuala Lumpur Malaysia (pp. 667-676).
- [27] Sakurai, Y., Yoshikawa, M., Faloutsos, C. (2005, June). FTW: fast similarity search under the time warping distance. In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems* (pp. 326-337).
- [28] Al-Naymat, G., Chawla, S., Taheri, J. (2012). Sparsedtw: A novel approach to speed up dynamic time warping. *arXiv preprint arXiv:1201.2969*.
- [29] Strle, B., Mozina, M., Bratko, I. (2009, June). Qualitative approximation to Dynamic Time Warping similarity between time series data. In *Proceedings of the Workshop on Qualitative Reasoning*.
- [30] Hsu, C. J., Huang, K. S., Yang, C. B., Guo, Y. P. (2015). Flexible dynamic time warping for time series classification. *Procedia Computer Science*, 51, 2838-2842.
- [31] Darabkh, K. A., Khalifeh, A. F., Bathech, B. A., Sabah, S. W. (2013, May). Efficient DTW-based speech recognition system for isolated words of arabic language. In *Proceedings of International Conference on Electrical and Computer Systems Engineering (ICECSE 2013)* (pp. 689-692).
- [32] Yadav, M., Aalam, A. (2016). Five Stage Dynamic Time Warping Algorithm for Speaker Dependent Isolated Word Recognition in Speech. *International Journal of Computer and Software Engineering (IJCSE)*, 4, 112-115.
- [33] Salvador, S., Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5), 561-580.
- [34] Mizutani, E. (2006, June). The dynamic time warping algorithms. In *Mechanical Engineering Seminar*, Tokyo Metropolitan University (p. III).
- [35] Shanker, A. P., Rajagopalan, A. N. (2007). Off-line signature verification using DTW. *Pattern recognition letters*, 28(12), 1407-1414.
- [36] Bahl, L., Jelinek, F. (1975). Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition. *IEEE Transactions on Information Theory*, 21(4), 404-411.
- [37] Sakoe, H., Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1), 43-49.
- [38] Jeong, Y. S., Jeong, M. K., Omitaomu, O. A. (2011). Weighted dynamic time warping for time series classification. *Pattern recognition*, 44(9), 2231-2240.
- [39] Górecki, T., Łuczak, M. (2013). Using derivatives in time series classification. *Data Mining and Knowledge Discovery*, 26(2), 310-331.
- [40] Zhao, J., Itti, L. (2018). shapeDTW: Shape dynamic time warping. *Pattern Recognition*, 74, 171-184.
- [41] Yuan, J., Lin, Q., Zhang, W., Wang, Z. (2019, November). Locally slope-based dynamic time warping for time series classification. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 1713-1722).
- [42] Lahreche, A., Boucheham, B. A Comparison Study of Dynamic Time Warping's Variants for Time Series Classification. *International Journal of Informatics and Applied Mathematics*, 4(1), 56-71.