

Full Paper

Handwriting recognition by derivative dynamic time warping methodology via sensor-based gesture recognition

Esra Tuncer and Mehmet Z. Unlu *

Izmir Institute of Technology, Department of Electrical and Electronics Engineering
35430 Urla, Izmir, Turkey

* Corresponding author, e-mail: zubeyirunlu@iyte.edu.tr

Received: 13 December 2021 / Accepted: 11 April 2022 / Published: 22 April 2022

Abstract: A handwritten character recognition methodology based on signals of acceleration obtained from gesture sensors with dynamic time warping (DTW) is presented. After applying the preprocessing steps of filtering, character separation and normalisation, similarities are detected by DTW and each signal component corresponding to a character is classified. However, the nature of the writing process may induce additional time-shifting problems among repetitions of characters since DTW uses only the amplitude values of signals to calculate the distance between them. Accordingly, when signals have different acceleration and deceleration values, irrelevant points of the signals may match each other just because their amplitude values are close. To overcome this problem, derivative dynamic time warping (DDTW) methodology is also implemented. The methodologies mentioned as well as the linear alignment approach were tested with Euclidean, Manhattan and Chessboard distance metrics to detect user-dependent/independent acceleration signals of lower-case characters of the English alphabets and digits. Recognition accuracy rates of Euclidean and Chessboard metrics with DDTW are 98.65%, which is the highest value among all methods applied and metrics. The comparison of Euclidean and Chessboard durations shows that Chessboard with DDTW is the most efficient method in terms of time.

Keywords: character recognition, three-axis accelerometer, dynamic time warping, derivative dynamic time warping

INTRODUCTION

Recognition of handwritten characters is a quite common studied subject in pattern recognition literature. Most of the methodologies are based on image processing and are called optical character recognition. The studies using different approaches are rather limited. One of these

approaches is called gesture-recognition-based character recognition that uses signals from body movements. The systems using this approach have varied application areas such as motion-to-text-to-speech for disabled people, recognition of sign languages and human-computer interaction. The new generation systems and tools developed for gesture-based interaction have provided new opportunities to interact with computers and other smart devices and have led to significant breakthroughs in human-computer interaction. For example, the use of touch screens in mobile phones has made it difficult for elderly or disabled individuals to use these devices, especially in the text-writing phase. However, with the help of a system developed in our study that can recognise characters, it will be very useful if they can physically draw the movement corresponding to that alphabet in the air or on a flat surface, and if these characters can be perceived and transferred to the electronic environment. The signals obtained by tracking hand movements can be acquired by imaging techniques or sensor-based techniques.

In sensor-based gesture recognition, researchers often use multi-axis accelerometers. An acceleration sensing glove that has accelerometers on the fingertips and back of a hand was introduced in the study of Perng et al. to detect simple hand gestures [1]. Using the effect of gravity, the glove operates like a computer mouse. The accelerometer placed on the upper back of the glove is utilised to move the pointer detecting the tilt motion and the remaining acceleration sensors help to work analogous to the click buttons. Another glove-like instrument constituted by Samsung Advanced Institute of Technology was introduced by Kim et al. [2]. A device named Magic Wand uses acceleration as well as angular velocity to gain extra knowledge about the movement. Using a similar device, the digits and three simple gestures were recognised by means of Bayesian networks and a 99.2% of recognition accuracy rate was achieved [3]. In another study a smartphone with a built-in accelerometer sensor was used. Klingmann [4] utilised five different gestures and a hidden Markov model for recognition and over 90% accuracy with ten training sequences was obtained.

To find similarities between signals shifted in the time domain, Akl and Valaee [5] used dynamic time warping (DTW) and affinity propagation methods together and a 94% accuracy rate was obtained with eighteen gestures and 3700 repetitions acquired from seven subjects. A comparison of Bayesian classification with feature separability weighting and DTW were introduced by employing only four different gestures with five repetitions from five individuals [6]. The characters were written in 3D space and the acceleration, angular velocity and orientations of the movement were used together. The gesture identification accuracies were 97% and 95% for the Bayesian classifier with feature separability weighting, and DTW respectively. In our previous study [7] by using similar gestures and methodology but with more characters and repetitions, a recognition accuracy rate of 98.08% was obtained for DTW.

In recent studies the number of work in which sensor-based gesture recognition methodology is used directly for character recognition is very few. In one of these studies Luo et al. [8] developed a wearable air-writing system in which users can type the English alphabets from A to Z in 3D space and the system recognises these characters using DTW with an inertial measurement unit. In this study the recognition accuracy was 84.6% for the upper-case alphabets in a user-dependent case. To verify signatures from which a time series can be obtained, as in the writing of characters, Stauffer et al. [9] proposed a graph-based approach to match the underlying signature graphs from different local perspectives and combined the resulting assignments by means of DTW and confirmed that the proposed methodology performs better concerning both the accuracy and run-time. In another recent study Cai et al. [10] proposed a novel time-series clustering approach named minimising dynamic time warping utilisation (MiniDTW) algorithm to

accelerate DTW and showed that MiniDTW, diminishing 98.52% of the DTW utilisation, is better than the counterpart method, TADPole, which only diminishes 75.56% of the DTW utilisation.

In this study a 3-axis accelerometer attached to a ring-like apparatus is employed to obtain the acceleration values of the handwriting movements on a flat surface parallel to the ground [11]. This kind of character recognition can be considered as gesture recognition, but its movements are more complex. Most researchers working on gesture recognition deal with fairly simple and basic gestures with a limited number of characters and numerals. Since our aim is the recognition of all 26 lower-case characters in the English alphabets (a-z) and digits (0-9) with minimum information about writing movement, 36 gestures with complex movements are collected by only using a 3-axis accelerometer. Then the acceleration signals are prepared for the recognition part. In addition to the use of conventional DTW with Euclidean distance, other distance metrics, Manhattan and Chessboard, are used in the recognition process, resulting in more efficient results. For the reasons that will be presented in the following pages, in addition to conventional DTW using raw data, derivative versions of the data giving shape information are also used. Results show that these new approaches proposed in this paper increase the recognition rate and efficiency of the system.

GESTURE-RECOGNITION-BASED CHARACTER RECOGNITION

A way to enable computers to recognise and interpret human body language is through gesture recognition. There are two main approaches in the literature for recognising gestures. In the first, where mainly image processing techniques are used, movements of body parts are captured using a camera and recognition is achieved in this way. In the second approach data describing the movements of body parts are obtained using a special set of sensors.

It can be thought that writing is a kind of gesture but with more complex movements. In this study, where character recognition is considered as gesture recognition, handwritten characters are recognised using data acquired from accelerometers that are mounted on a ring-like device. Single-axis or multi-axis accelerometers can be used to acquire data consisting of acceleration values associated with gesture information at the time of writing. In order to understand what acceleration data are and how they are acquired, it would be useful to review shortly the terms *position*, *displacement*, *velocity*, and *acceleration of motion* first.

An object's position at any particular time is the place where that object is located. To describe the motion of an object, its exact position relative to a reference point must be specified. The reference points are not necessarily to be stationary but can also be in motion, but generally the Earth is used as the reference point. Three variables x , y and z are used to describe the position of an object in three dimensions. If we consider a one-dimensional displacement, the displacement in the x direction, Δx , is the change of position of an object relative to a reference point, x_0 , and is defined as

$$\Delta x = x_f - x_0 \quad (1)$$

where x_f is the final position and x_0 is the initial position.

The velocity of an object is the rate at which its position changes over time. So the average velocity, v_{avg} , in the x direction is defined as

$$v_{\text{avg}} = \frac{\Delta x}{\Delta t} = \frac{x_f - x_0}{t_f - t_0} \quad (2)$$

where Δt represents the change in time, t_f and t_0 are the final and the initial time instants respectively. Equation (2) implies that the velocity in any direction is equal to the slope of a

position-time graph. Considering the position-time graph in Figure 1 as an example, the slope values for each time instant can be obtained using equation (2).

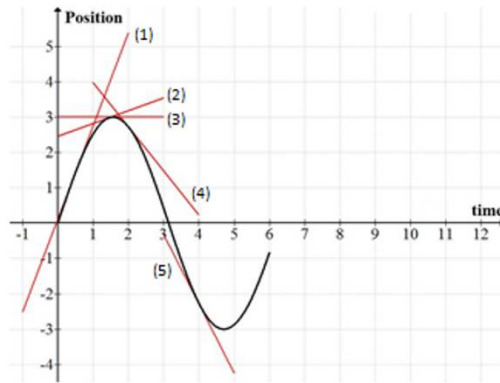


Figure 1. Position vs time graph of an object

Here, the red lines are examples of slopes at particular time instants. As in slopes (1) and (2), the slope, corresponding to velocity, has positive values from the beginning of the graph to the first peak indicating that the object is moving in a positive direction. Similarly, but conversely, as with slopes (4) and (5), the slope has negative values indicating that the object is moving in a negative direction. When there is no displacement for the object, then the slope and the velocity are zero as in (3).

The non-linearity of the position graph indicates that the slope changes and hence the velocity changes with time. Acceleration is caused by a change in velocity. Accordingly, the non-linearity of the position-time graph means that the object is accelerating or decelerating and thus has acceleration. The acceleration, a , can be defined as the rate of change of its velocity in time as in equation (3).

$$a = \frac{\Delta v}{\Delta t} = \frac{v_f - v_i}{\Delta t} \quad (3)$$

Measuring Acceleration

Electronic sensors used to measure the acceleration of an object are called accelerometers. Even if there is no movement, an accelerometer produces an output as gravitational acceleration, g , pointing directly towards the sphere's centre in the amount of 9.81 m/s^2 at sea level. Accelerometers, which are available in capacitors, piezoresistors, magnetoresistors, piezoelectric, and Hall effect types, are commonly used in industry as in navigation systems for aircrafts and missiles and in most of electronic devices such as drones, smartphones and tablet computers. To understand how an accelerometer works and how it produces an output, consider a ball inside a structure and the x and z axes and their opposite directions, as shown in Figure 2.

In a condition in Figure 2(a), when there is no movement in any direction and no gravity effect, the ball hangs in the air. Hence the accelerometer produces $0g$ in both axes as its output. If the two-axis structure moves to the left with an acceleration of $1g$ as in Figure 2(b), the ball moves towards the $+x$ side with an acceleration of $1g$. Accordingly, $1g$ and $0g$ in the x - and z -axes respectively will be acquired as the accelerometer output. In the case of Figure 2(c), there is a gravitational force of $1g$ towards $-z$ direction, in which case the ball will be positioned at the lower edge of the structure towards $-z$ -axis, causing the output of $-1g$ and $0g$ at the accelerometer along

the z - and x -axes respectively. For the last case shown in Figure 2(d), there are both the gravitational force towards $-z$ direction and movement to the $-x$ direction. Then the ball will be positioned at the bottom right corner of the structure, and so the output of the accelerometer will produce $1g$ towards both $+x$ and $-z$ directions.

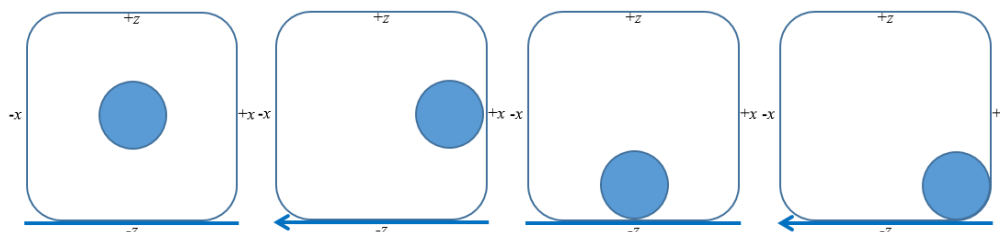


Figure 2. Four different cases of the ball inside the structure: (a) no gravitational force, no movement, (b) no gravitational force, structure moving towards left direction, (c) gravitational force of $1g$, no movement, (d) gravitational force of $1g$, structure moving towards left direction

Figure 3 shows examples of 3-axis raw acceleration signals obtained in the study for the lower-case letters ‘a’, ‘b’ and ‘c’. There is no significant difference among acceleration values on the z -axis since the writing process takes place only in x and y planes. The graphs mainly include acceleration caused by hand movements in x and y directions. The acceleration on the z -axis constitutes only quite small variations around $-1g$ compared to the other directions.

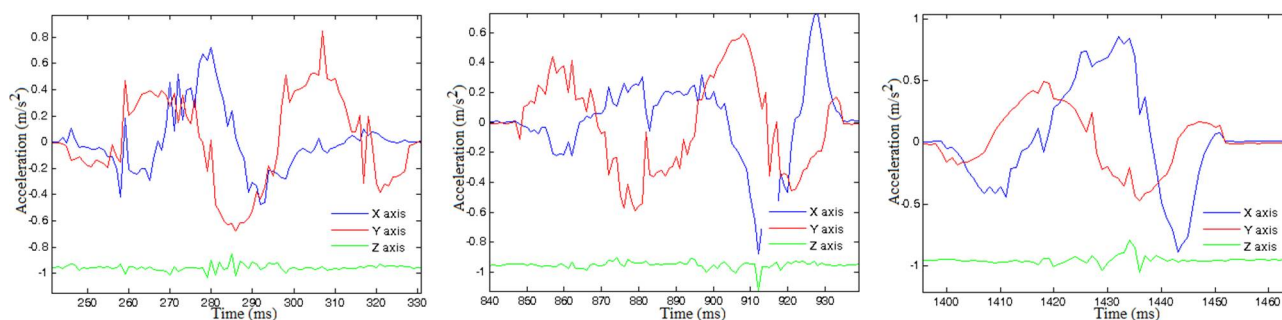


Figure 3. Raw data signals in x , y and z directions for lower-case characters ‘a’, ‘b’ and ‘c’ respectively

DERIVATIVE DYNAMIC TIME WARPING METHODOLOGY FOR CHARACTER RECOGNITION

DTW is a technique utilised to provide an optimal pairing between two-time series with certain constraints and regulations by calculating the distance or similarity between them. However, in DTW similarity comparison between two signals is not carried out using simultaneous amplitude points as in linear alignment. Instead, a sample of the considered signal is compared with all samples of the other signal, and this process is repeated for all other samples of the first signal, yielding a distance matrix that shows the distances among all samples of both signals. Acquisition of the distance matrix yields the most suitable path, named warping path, and the two signals may be aligned in time by warping according to this path. This sequence alignment approach can be employed to classify the time series. Unlike DTW, the derivative dynamic time warping (DDTW) method performs this alignment process by using the shape information obtained by taking the first-order derivatives of the signals.

To obtain optimum results from both DTW and DDTW, the signals must be preprocessed. For this reason, firstly the applied preprocessing steps are explained, and then the details of DTW and DDTW implementations are given.

Preprocessing

For more accurate and reliable recognition results, the data acquired from the acceleration sensor should be enhanced first. For this purpose, three preprocessing steps shown in Figure 4 are applied to the signals. The first step is filtering. After filtering the signal, the functional parts of the signal corresponding to the characters need to be segmented out from the parts where there is no character information. Finally, a normalisation process is applied to the signals segmented to bring all character signals to the same amplitude range.

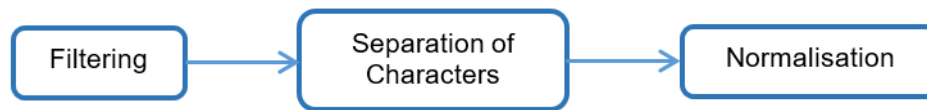


Figure 4. Sequential processes applied in preprocessing step

Filtering

A moving average filter (MAF) is used to eliminate the inner noise sourced by the acceleration sensor and the noise caused by hand vibrations. The MAF is a low-pass filter and its cut-off frequency can be adjusted by changing its length.

Figure 5 shows an example of the frequency spectrum of the raw data sample. The spectrum shows that the signal components carrying functional information are mostly between 0 - 4 Hz and the remaining frequency components correspond to high-frequency noise. The MAF averages the specified number of the signal instances by means of equation (4),

$$y[n] = \frac{1}{1 + (M_2 - M_1)} \sum_{k=-M_1}^{M_2} x[n-k] \quad (4)$$

where x and y are the input and output signals respectively, n specifies the sample on the signal, k mentions the delay amount in time, and M_1 and M_2 define the window size.

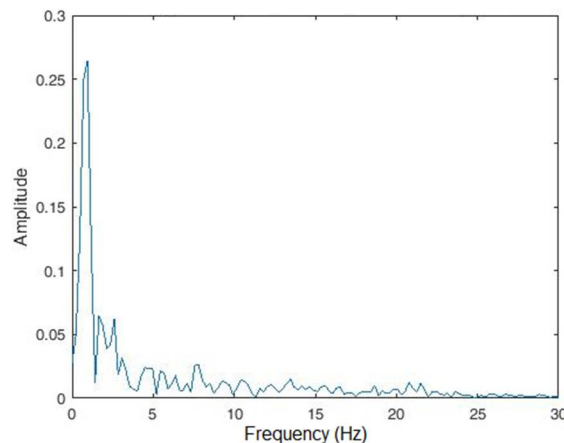


Figure 5. Typical raw acceleration signal magnitude spectrum

Considering the frequency response of MAFs of different sizes, given in Figure 6 and obtained by equation (5), the optimal filter length for eliminating frequency components higher than 4 Hz appears to be 15.

$$H(e^{j\omega}) = \frac{1}{M_1 + M_2 + 1} \frac{e^{j\omega M_1} - e^{-j\omega(M_2+1)}}{1 - e^{-j\omega}}. \quad (5)$$

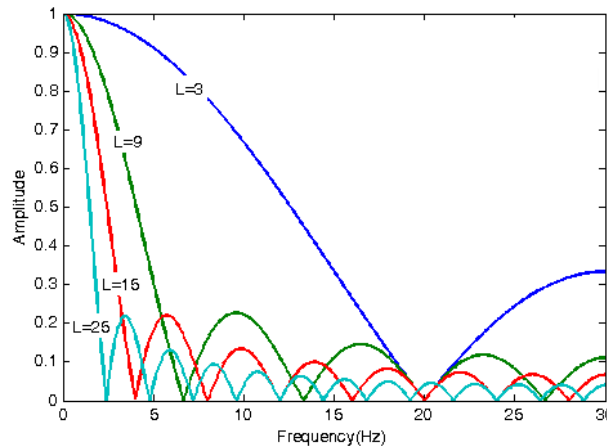


Figure 6. Comparison of frequency response of several MAFs of different lengths (L)

Separation of characters

The next step after maintaining smoother signals through filtering is the segmentation of signal parts that contain functional information about the writing process for each character. Here, two different segmentation approaches are used. In the first the template signals to be used in the recognition stage, and in the other, the characters of the words written to be recognised are segmented.

The acquisition of the template signals is carried out continuously by collecting and recording repetitions of each character in the same signal sequence. Each sequence contains 20 repetitions of that character to be used separately in the recognition part. As shown in Figure 7, there are parts of the sequence where there is no movement or acceleration, the amplitude values being close to zero, and there is no information about the characters, which is caused by the users waiting about three seconds between each repetition of a character.

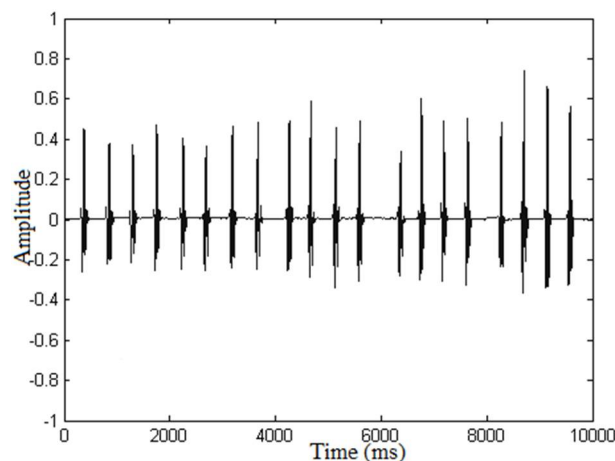


Figure 7. Twenty filtered repetitions of the lower-case 'a' character

To segment out each repetition, the first and last instances of each character must be detected. Large variations in amplitude values occurring within a certain range can be used to find these points. To observe the changes in amplitude, consider a variable k . If there is a change between the current instance $x[i]$ and the next one $x[i+1]$ and this change is greater than a specified threshold value m , then a specified t value is added to the variable k representing the change. The difference in variation between data instances is positive or negative, but the absolute value of the difference can be used. If the difference is not greater than a determined value of m , it means that there is no change in amplitude. However, deciding the value of m is challenging. If the value assigned is too large, it is possible to lose parts containing character information, and if it is too small, it is possible to add vibrations in the parts corresponding to the waiting parts in the signal. The optimum difference value threshold for segmenting each character was found to be 0.002 to be used in the study as in Figure 8. Figure 9 shows an outcome of the segmentation process.

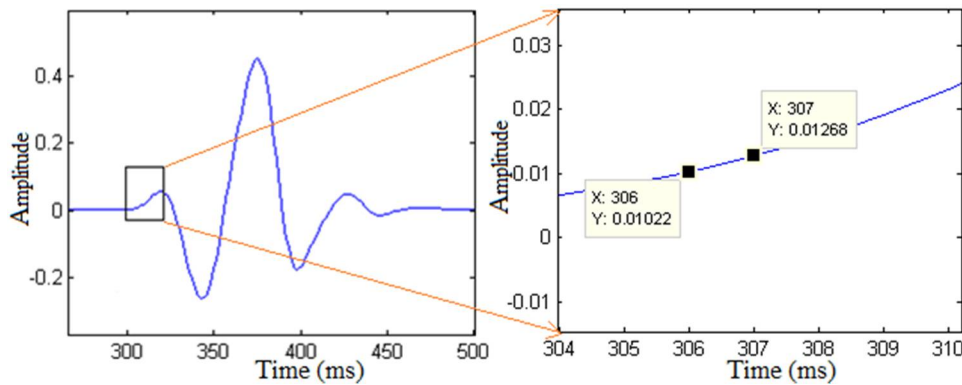


Figure 8. Amplitude change between characters

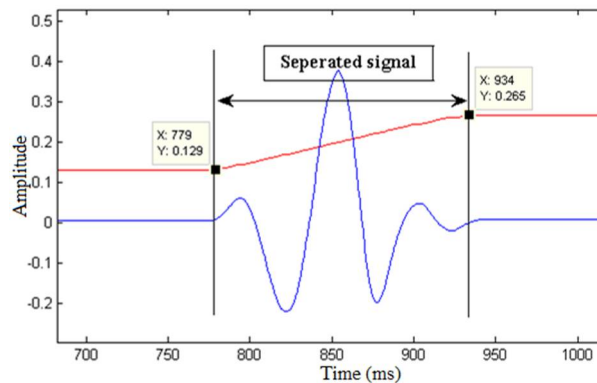


Figure 9. Detecting initial and final instances of a character acceleration signal. Sloping red line emphasises variation on k variable in amplitude.

When there is a large oscillation from which changes in amplitude value between characters are greater than m , then the estimated length of the acceleration signal for the character is used to overcome this problem. Character signals generally appear to be longer when their lengths are compared to pure vibration lengths. Therefore, the minimum possible character signal length is chosen to include 50 data points. In this study three different data are acquired for each character from the 3-axis accelerometer. However, only x - and y -axis data are taken into account as there is no significant movement in the z -axis at the time of writing. Sometimes, data obtained from an axis during writing occupy more space than that by the other axis. In this case when only one axis is

considered in finding the separation points, important parts of the other axis will likely be lost. Therefore, possible initial and final instances of a single character repeat on both axes are found first, and if one of the sequences contains more samples than the other, the first and last data instances are determined from that one.

To segment out characters in written words during the creation of the database, a slightly different approach is used; the difference is the axes used. When a text is written, users write each character in x - and y -axes. After each character is written, there is a movement out of the writing plane creating an acceleration on the z -axis before writing another character. This movement, causing acceleration in the z -axis between each character, is used to segment out characters within texts.

Normalisation

The signals acquired from an accelerometer might be of different lengths and different amplitude ranges even for the same character repetition, as shown in Figure 10. Even if the same character is written twenty times by a person, the acceleration and deceleration among repetitions will be different. Signals belonging to a character have similar overall shapes but different amplitudes and lengths. The DTW methodology, which uses amplitude values to match signals, may overcome the length-inequality difficulty. However, the difference in amplitude values may cause mismatching between data samples and result in inaccurate recognition. Therefore, before moving on to the next steps, all segmented signals in the data set must be scaled within the same amplitude limits. Since raw acceleration data can have both negative and positive values, the signals are normalised between -1 and +1 using the normalisation equation (6),

$$y = -1 + (x - A) \cdot (1 - (-1)) / (B - A) \quad (6)$$

where x is the signal amplitude value to be normalised, A and B are the highest and lowest amplitude rates of the signal respectively, and y is the normalised signal value.

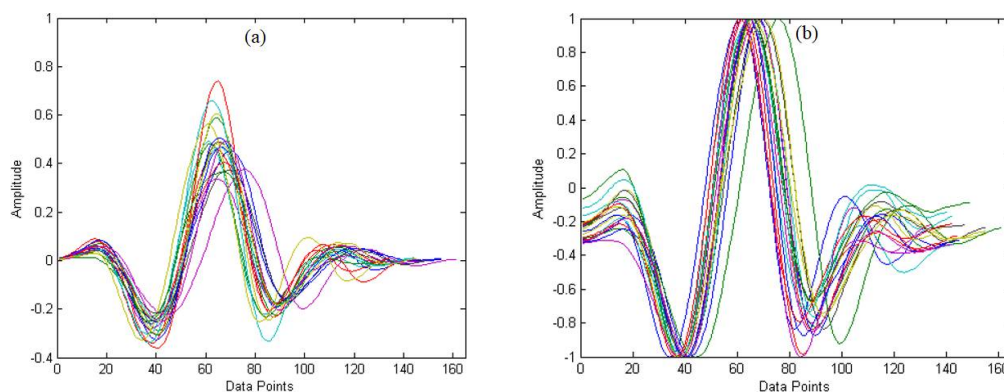


Figure 10. Acceleration signals obtained by repeating twenty times for the same character before normalisation (a) and after (b)

DTW Methodology

In a basic linear similarity calculation the simultaneous amplitude points of the signals are used to perform the distance measurement. The difference in length between the two time series makes it impossible to calculate the distance with linear alignment. Besides, if the signal parts to be aligned are not in the same time interval, the linear alignment results will also be incorrect. In the

case that there are different acceleration and deceleration amplitude values between two signals, the linear similarity result might be wrong. Figure 11 shows an alignment with linear similarity calculation and a true alignment with a non-linear similarity calculation.

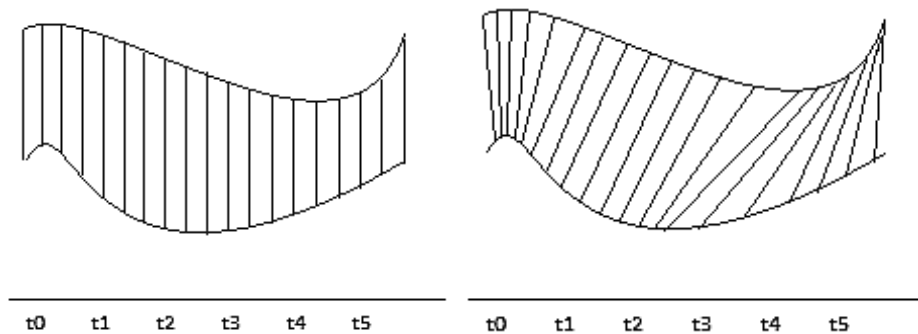


Figure 11. Linear (left) and non-linear (right) alignments of signals having different corresponding acceleration amplitude values

In DTW, a sample of the considered signal is compared with all samples of the other signal by calculating distances to obtain the true alignment with non-linear similarity calculations. By applying the same process to all other samples of the first signal, a complete distance matrix is obtained. Finding the optimal matching on the distance matrix gives the similarity measurement. Euclidean distance matrix elements for the signals s and t can be calculated via the equation,

$$d_{\text{euc } ij} = \sqrt{(s_i - t_j)^2} \quad (7)$$

where i and j are the instance numbers of the signals. An example of a distance matrix calculated for s and t signals is given in Figure 12.

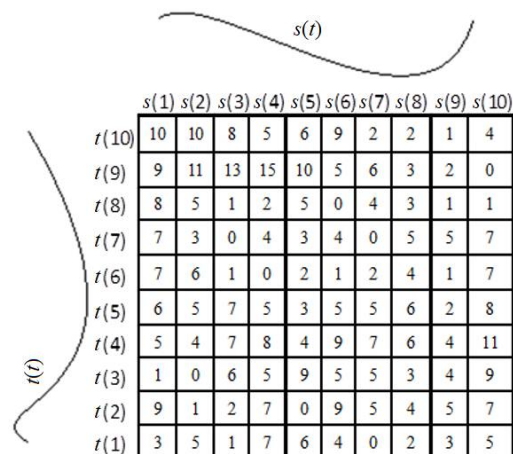


Figure 12. Example of a distance matrix calculated for s and t signals. Each matrix element represents the distance between corresponding data samples $s(i)$ and $t(j)$.

A 3-axis accelerometer is used to collect data from the writing process of 26 lower-case characters of the English alphabets and 10 digits. So there are three distinct data sequences in the x -, y -, and z -axes for each character and digit in the data set. However, since writing is performed in the plane parallel to the ground, there will be no significant formation in the z -axis. Therefore, to construct the distance matrix in Figure 12, just the x - and y -axis data are utilised in conjunction with the 2D Euclidean distance obtained by equation (8):

$$d_{2D \text{ euc } ij} = \sqrt{(s_{x_i} - t_{x_j})^2 + (s_{y_i} - t_{y_j})^2} \quad (8)$$

In conventional DTW applications in the literature, although the square and square root operations increase the computation time, the Euclidean distance equation is generally used. To speed up distance calculations, other distance measurement approaches that are not used with DTW in the literature such as Manhattan and Chessboard are also applied and their performances are compared.

The 2D Manhattan distance equation to create the distance matrix can be defined as

$$d_{\text{manh } ij} = |s_{x_i} - t_{x_j}| + |s_{y_i} - t_{y_j}|, \quad (9)$$

and similarly the 2D Chessboard distance equation is defined as

$$d_{\text{chess } ij} = \max(|s_{x_i} - t_{x_j}|, |s_{y_i} - t_{y_j}|). \quad (10)$$

After the creation of the distance matrix, the next step in DTW is finding the optimal path on the matrix that can give the best true alignment between two signals. There may be thousands of possible paths in a distance matrix. To eliminate them and to find the optimal one, some restrictions need to be used. Assuming that the signals s and t have the length of L_s and L_t respectively, their warping path (W) with the length of K is defined as

$$W = w_1, w_2, \dots, w_k, \dots, w_K, \max(L_t, L_s) \leq K < L_s + L_t - 1. \quad (11)$$

Then the restrictions are as follows [12,13]:

Boundary Conditions. Initial point w_1 and ending point w_K of the warping path defined in equation (11) must be located at the diagonal corners of the matrix:

$$w_1 = (1,1), \text{ then } w_K = (L_t, L_s).$$

Continuity. The following instance of the warping path should not be more than a cell ahead:

$$w_{k+1} = (a', b'), w_k = (a, b), \text{ then } a' - a \leq 1, \quad b' - b \leq 1.$$

Monotonicity. The warping path needs to start from a corner and continue monotonically to the counter corner to prevent re-matching problems:

$$w_{k+1} = (a', b'), w_k = (a, b), \text{ then } a' \geq a, \quad b' \geq b.$$

The restrictions mentioned above limit the search area for the optimal path, but there still are many possible paths available. To achieve more meaningful constraints, consider the warping-cost concept. Warping-cost is a cumulative summation of distance value for each path. A path that gives the minimum cost can be chosen as an optimal path since the corresponding distance of the chosen path is minimum and the similarity is maximum. When n possible warping paths are taken into account, the lowest cost is defined as

$$DTW_{\text{cost}} = \min \left(\sum_{k=1}^K W_k \right). \quad (12)$$

The optimal path that reduces the warping cost most can be discovered using dynamic programming algorithm [5, 14]:

$$D(i, j) = d(s_i, t_j) + \min \begin{pmatrix} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{pmatrix} \tag{13}$$

where $D(i, j)$ is the value of a cell in the distance matrix and the distance function $d(s_i, t_j)$ is the cost function of two elements s_i and t_j which are the i^{th} and j^{th} samples of s and t signals respectively.

In Figure 13 some of the possible warping paths are shown in different colours. By using the dynamic programming equation (13), the optimal path is found. For the example given in Figure 13, the purple path is the optimal path for minimising the warping cost.

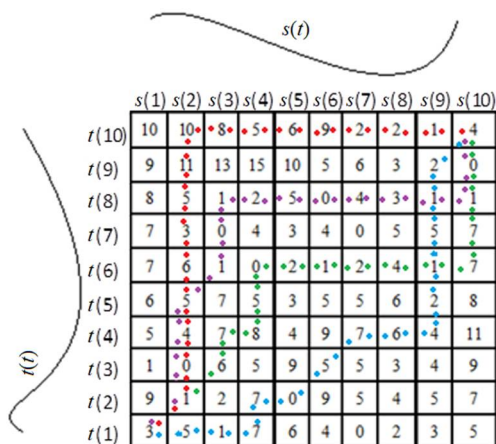


Figure 13. Possible warping paths shown in different colours

To avoid mismatching, the warping path must be around the diagonal. In some cases the amplitude rates of the distinct portions of the signals may be closer than the portions of the signal anticipated to match. In such cases the path may move away from the diagonal line. To prevent this problem, a warping window is used. The warping window reduces the search area and distance measurement is solely performed for the cells corresponding to the window, thus speeding up the DTW algorithm. Any value from the length difference of the two signals to the full size of the matrix can be selected as the window size. Figure 14 represents how the warping window reduces the search area and eliminates non-optimal paths. According to this, the red path is the optimum one and the purple path is the one that is eliminated employing the warping window.

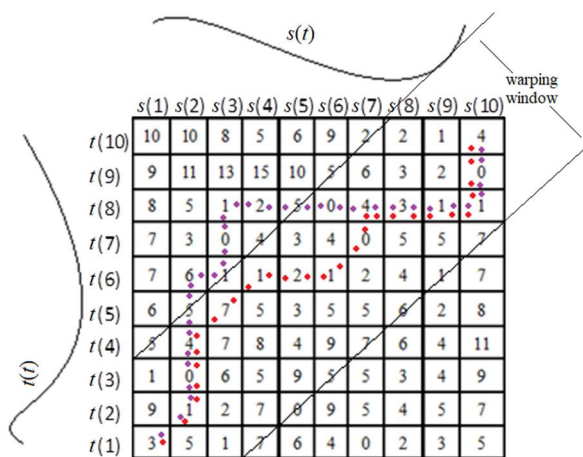


Figure 14. Warping paths constrained by a window

Singularity Problem and Solution via DDTW

As mentioned above, the time series even from the same character may differ from each other on the time axis. DTW is the method to overcome this problem. However, the time series may also differ on the amplitude axis. This is another problem that needs to be solved for DTW. Because DTW only uses amplitude values of the signals to find similarities, this problem is not solved by the restrictions mentioned above. As a result, some unrelated signal portions can be paired because their amplitude rates are closer to each other than the part that needs to be matched, or a large piece of a signal may be paired with a single sample of the other signal (Figure 15). Both of these problems are called singularity problem [15].

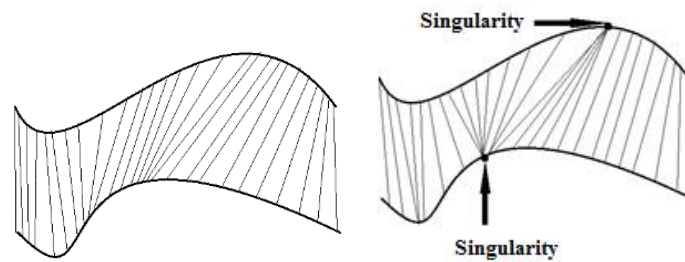


Figure 15. True matching and matching with singularities

Keogh and Pazzani [12] proposed a solution to the singularity problem with a method called DDTW. In this method the shape information of the signals is extracted and used with DTW instead of using only the low-pass filtered raw signal. The use of shape information acquired by taking the first derivative reduces the effect of mismatching caused by amplitude differences. The following equation is used for this purpose:

$$\frac{dq}{di} = \frac{(q_i - q_{i-1}) + ((q_{i+1} - q_{i-1}) / 2)}{2}, \quad (14)$$

where q is the signal to be derived and i is the instance number. The conventional derivation equation (15) can also be used for the same purpose:

$$\frac{dq}{di} = q_i - q_{i-1}. \quad (15)$$

Both derivation equations lead to close accurate recognition rates. To see the effect of derivation particularly, the normalisation step that expands and limits the amplitude interval of the signals between -1 and +1 is removed just for the derivational part.

EXPERIMENTAL RESULTS AND ANALYSIS

A data acquisition system containing an MPU-6050 accelerometer, an ARM mbed NXP LPC1768 development board and a micro SD card module was built and used to acquire the data. Although the accelerometer has a 3-axis gyroscope ranges of ± 250 , ± 500 , ± 1000 and $\pm 2000^\circ/\text{sec}$ (dps) and a 3-axis accelerometer with $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$ ranges, only $\pm 2g$ acceleration was used.

Using the system designed, two types of data set, user-dependent and user-independent, were acquired to determine comprehensively which methodology gives the highest recognition rate in what processing time. User-dependent data were derived from the acceleration data of a single user who knows well how to use the ring-like data acquisition apparatus. User-independent data, not

used as template data, were collected from five distinct users using the ring-like apparatus for the first time.

The experimental results and analysis were divided into two parts: recognition and classification of user-independent data using a user-dependent template, and recognition and classification of user-dependent data using a user-dependent template. The second part was also divided into two within itself as raw data results and derivative data results.

First, each data sample in the user-independent data set was cross-checked with the signals in the database containing user-dependent data, and the signal selected to be compared was labelled with the signal providing the minimum total distance using linear alignment as well as DTW methodology. Figure 16 shows the recognition accuracy rates and processing times of this evaluation when Euclidean, Manhattan and Chessboard distance metrics are used. It should be noted here that the durations mentioned are those of comparing all database repetitions with each other. Therefore, it should be considered normal that it takes a lot of time, especially when using DTW. As presented in the figures, the results show the effectiveness of DTW over the linear alignment. The recognition accuracy rate increases by about 10% for each distance metric when DTW is used. When the metrics are compared, DTW with Chessboard is the most effective one giving the highest recognition rate (69.23%) with minimum duration.

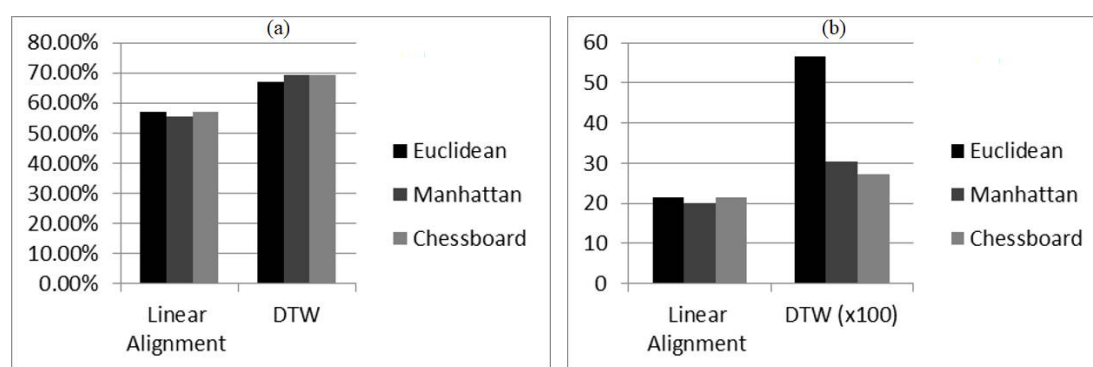


Figure 16. Recognition accuracy rates (a) and respective processing times in seconds (b) when user-independent data are labelled by means of user-dependent template data

Results of the rest of the experimental studies are given by dividing user-dependent data results into two parts: raw data and derivative data results. Here, the processing time is a critical point for DTW since the leave-one-out approach is used, which means selecting one sample for comparison with all other samples to obtain the results.

Figure 17 shows that, for the recognition and classification of user-dependent raw data using a user-dependent template, DTW when applied with Manhattan distance is the most effective one, giving the highest recognition rate of 97.88% with minimum duration. However, using signal derivatives instead of raw signals enhances the recognition rate. Figure 18 shows that DDTW with Chessboard is the most effective one and gives the highest recognition rate of 98.65% with minimum duration. To see the real effect of derivation, the normalisation step that expands and limits the amplitude range of the signals between -1 and +1 is removed from preprocessing part of the DDTW application. Figure 19 compares the overall recognition rates and time durations of DTW and DDTW for user-dependent data with three different distance metrics. Figure 19 shows that DDTW with Euclidean and Chessboard achieves the highest recognition rate of 98.65%, whereas DDTW with Chessboard gives minimum time between the two.

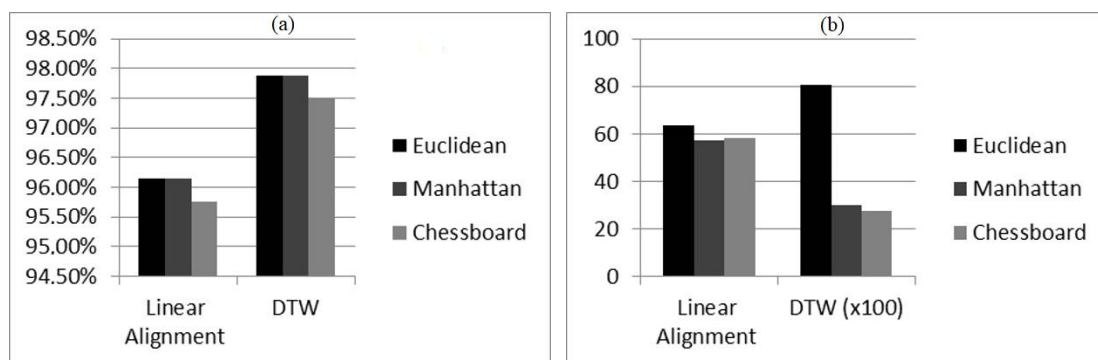


Figure 17. Recognition accuracy rates (a) and processing times in seconds (b) for raw user-dependent data

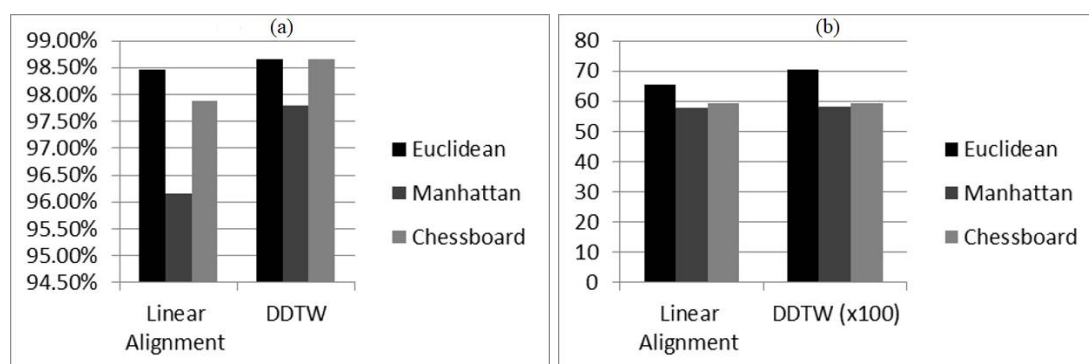


Figure 18. Recognition accuracy rates (a) and processing times in seconds (b) for derivative user-dependent data

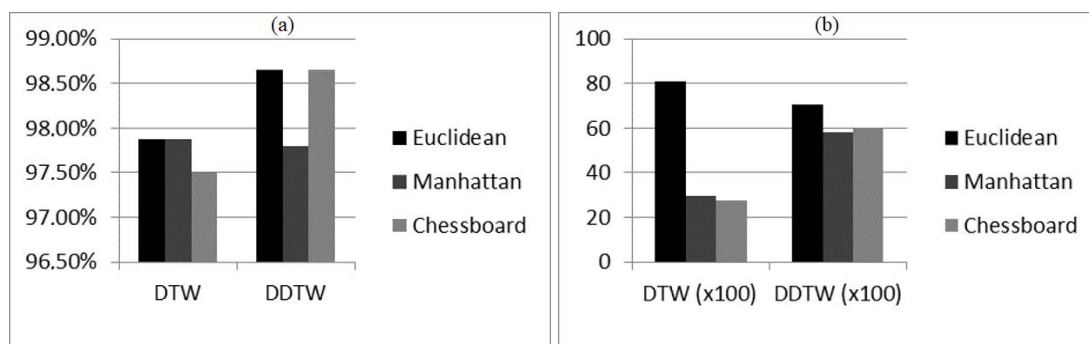


Figure 19. Comparison of overall recognition accuracy rates (a) and processing times in seconds (b) of DTW and DDTW

Figure 20 compares the best of the results of our methodology with similar studies in the literature. The studies of Choi et al. [16] and Jeon-Shing et al. [17] involve only numerical digits (0-9). Our study and that of Patil et al. [18] include both the digits and lower-case English characters (0-9, a-z). Our first approach, DTW with Manhattan distance on raw data, results in 97.88% accuracy. While Patil et al. [18] achieve a recognition rate of 97.95%, our second approach, DDTW with Chessboard distance gives 98.65% accuracy.

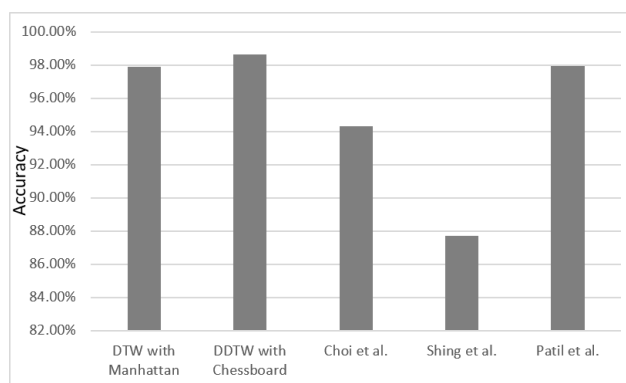


Figure 20. Overall comparison of results in the study and those in literature

CONCLUSIONS

The overall results show that DTW gives improved results than linear alignment because the acceleration signal samples are also shifted by different amounts over time. On the other hand, by considering the results in Figure 18, using the derivative versions of the signals (DDTW) can solve the singularity problem and increase the recognition accuracy rates. The normalisation step in DDTW is removed, indicating that the derivation process brings up the shape information of the signals and DDTW is more effective than DTW. The approach using Manhattan or Chessboard distance metrics with conventional DTW reaches the accuracy rate of Euclidean metric for both raw and derivative data and also reduces the processing time significantly. Although these distance measurements can reduce the time duration, it is still a major problem for real-time systems since processing a single character takes roughly 5-6 seconds. Some approaches such as reducing the resolution or quantising the signals during the simulation process were tested to speed up the system, but it was concluded that they caused a decrease in recognition rates.

REFERENCES

1. J. K. Perng, B. Fisher, S. Hollar and K. S. J. Pister, "Acceleration sensing glove (ASG)", Proceedings of 3rd International Symposium on Wearable Computers, **1999**, San Francisco, USA, pp.178-180.
2. Y. S. Kim, B. S. Soh and S. G. Lee, "A new wearable input device: SCURRY", *IEEE Trans. Indus. Electron.*, **2005**, 52, 1490-1499.
3. S. J. Cho, J. K. Oh, W. C. Bang, W. Chang, E. Choi, Y. Jing, J. Cho and D. Y. Kim, "Magic wand: A hand-drawn gesture input device in 3-D space with inertial sensors", Proceedings of 9th International Workshop on Frontiers in Handwriting Recognition, **2004**, Tokyo, Japan, pp. 106-111.
4. M. Klingmann, "Accelerometer-based gesture recognition with the iPhone", *MSc Thesis*, **2009**, Goldsmiths University of London, UK.
5. A. Akl and S. Valae, "Accelerometer-based gesture recognition via dynamic time warping, affinity propagation, and compressive sensing", Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, **2010**, Dallas, USA, pp.2270-2273.
6. D. Mace, W. Gao and A. Coskun, "Accelerometer-based hand gesture recognition using feature weighted naïve Bayesian classifiers and dynamic time warping", Proceedings of Companion Publication of 2013 International Conference on Intelligent User Interfaces Companion, **2013**, Santa Monica, USA, pp.83-84.

7. E. Tunçer and M. Z. Ünlü, "Accelerometer based handwritten character recognition using dynamic time warping method", Proceedings of 24th Signal Processing and Communication Application Conference, **2016**, Zonguldak, Turkey, pp.1805-1808.
8. Y. Luo, J. Liu and S. Shimamoto, "Wearable air-writing recognition system employing dynamic time warping", Proceedings of IEEE 18th Annual Consumer Communications and Networking Conference, **2021**, Las Vegas, USA, pp.1-6.
9. M. Stauffer, P. Maergner, A. Fischer, R. Ingold and K. Riesen, "Offline signature verification using structural dynamic time warping", Proceedings of International Conference on Document Analysis and Recognition, **2019**, Sydney, Australia, pp.1117-1124.
10. B. Cai, G. Huang, N. Samadiani, G. Li and C. Chi, "Efficient time series clustering by minimizing dynamic time warping utilization", *IEEE Access*, **2021**, 9, 46589-46599.
11. E. Tunçer, "Accelerometer based handwritten character recognition using dynamic time warping", *MSc Thesis*, **2016**, İzmir Institute of Technology, Turkey.
12. E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping", Proceedings of 1st SIAM International Conference on Data Mining, **2001**, Chicago, USA, pp.1-11.
13. L. Yousofvand, A. Fathi and F. Abdali-Mohammadi, "Person identification using ECG signal's symbolic representation and dynamic time warping adaptation", *Signal Image Video Process.*, **2019**, 13, 245-251.
14. Q. Chen, G. Hu, F. Gu and P. Xiang, "Learning optimal warping window size of DTW for time series classification", Proceedings of 11th International Conference on Information Science, Signal Processing and their Applications, **2012**, Montreal, Canada, pp.1272-1277.
15. S. L. Culp, "Warping methods for means and variances in functional data", *PhD Thesis*, **2008**, University of Michigan, USA.
16. S. Choi, A. S. Lee and S. Lee, "On-line handwritten character recognition with 3D accelerometer", Proceedings of IEEE International Conference on Information Acquisition, **2006**, Weihai, China, pp.845-850.
17. J. S. Wang, Y.-L. Hsu and C. L. Chu, "Online handwriting recognition using an accelerometer-based pen device", Proceedings of 2nd International Conference on Advances in Computer Science and Engineering, **2013**, Los Angeles, USA, pp.229-232.
18. S. Patil, D. Kim, S. Park and Y. Chai, "Handwriting recognition in free space using WIMU-based hand motion analysis", *J. Sensors*, **2016**, 2016, Art.no.3692876.