

CHAPTER III

SHAPE-BASED AVERAGING

Since the causes of having cluster representatives, the outputs generated from a Subsequence Time Series Clustering (STSC) with both k -means clustering and k -hierarchical clustering, becoming all sine waves are inappropriate uses of Euclidean distance and Amplitude Averaging as a distance measure and an averaging function, respectively, in this chapter, Shape-based Averaging is proposed to use instead of Amplitude Averaging in STSC to correctly generate a cluster representative from trivial-matched subsequences. Unlike other typical data types, time series data need Shape-based Averaging instead of Amplitude Averaging since correlations among adjacent dimensions exist (Niennattrakul and Ratanamahatana, 2007a,b). Additionally, Amplitude Averaging produces an undesired mean, where this leads to an inaccurate cluster representative. Figure 3.1 shows the results from averaging of two time series sequences A and B using Amplitude Averaging and Shape-based Averaging, respectively. The sequence generated from the Amplitude Averaging shows an undesired averaged result that contains two events, where both original sequences A and B consist of only one event. The sequence generated from Shape-based averaging preserves characteristics of these two data sequences that only one event exists.

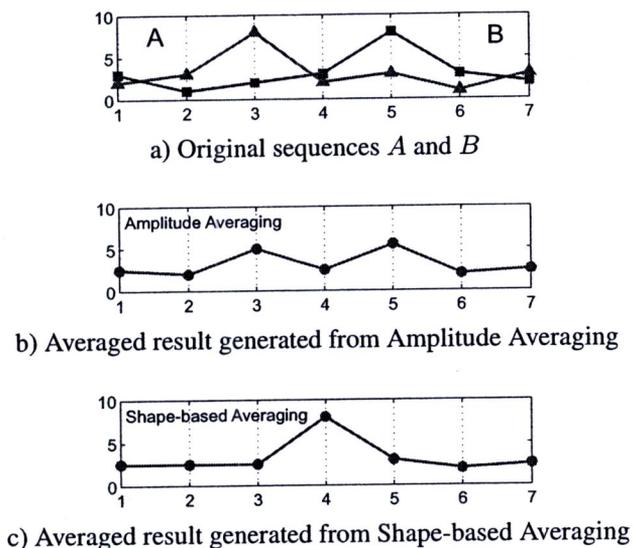


Figure 3.1: Comparison between two averaged results generated from Amplitude Averaging and Shape-based Averaging.

However, constructing an accurate shape-based mean is still controversial because data sequences are averaged in Dynamic Time Warping (DTW) distance space not in the Euclidean space. Unfortunately, no optimal solution has been proposed. Over a decade ago, Gupta et al.

proposed a heuristic solution called NLAAF (Gupta et al., 1996), while only a handful number of work has been adapted to time series data mining domain (Ratanamahatana and Keogh, 2005a; Salvador and Chan, 2007). Particularly, NLAAF does not produce good averaged results since an averaged result is always longer than the original sequence and has large errors. In this thesis, a new averaging scheme with two averaging functions, Cubic-Spline Dynamic Time Warping (CDTW) averaging and Iterative Cubic-Spline Dynamic Time Warping (ICDTW) averaging, is introduced. With the proposed construction algorithm, a very well-formed mean is generated. Averaged results generated from NLAAF, CDTW, and ICDTW are compared and evaluated in terms of SUMDIST, a summation of distances between the averaged result and all original sequences.

3.1 Background

This section provides essential background knowledge, i.e., Dynamic Time Warping (DTW) distance and Dynamic Time Warping (DTW) averaging function, to understand proposed methods in this chapter.

3.1.1 Dynamic Time Warping (DTW) Distance

DTW distance (Berndt and Clifford, 1994; Ratanamahatana and Keogh, 2005b) is a well-known shape-based similarity measure that uses a dynamic programming technique to find an optimal warping path between two time series sequences. To calculate the distance, it first creates a distance matrix, where each element in the matrix is a cumulative distance of the minimum value of three surrounding neighbors. Given two time series sequences $A = \langle a_1, a_2, \dots, a_i, \dots, a_n \rangle$ and $B = \langle b_1, b_2, \dots, b_j, \dots, b_m \rangle$, an n -by- m matrix is first created, and then each (i, j) element $\gamma_{i,j}$ of the matrix is defined as:

$$\gamma_{i,j} = |a_i - b_j|^p + \min \{ \gamma_{i-1,j-1}, \gamma_{i-1,j}, \gamma_{i,j-1} \} \quad (3.1)$$

where $\gamma_{i,j}$ is the summation of $|a_i - b_j|^p$ and the minimum cumulative distance of three elements surrounding the (i, j) element, and p is the dimension of L_p -norms. When all elements in the matrix are filled, DTW distance is determined from the last element $\gamma_{n,m}$ of the matrix. For time series domain, $p = 2$, equipping to Euclidean distance, is typically used. Since DTW distance is important background knowledge for this thesis, a pseudo code is provided in Table 3.1 and an illustrative example of DTW distance calculation is shown in Figure 3.2.

Table 3.1: Pseudo code of Dynamic Time Warping distance measure

FUNCTION $[dist] = DTW-DISTANCE [A, B]$	
1.	Let n be the length of time series A
2.	Let m be the length of time series B
3.	Let p be the dimension of L_p -norms
4.	Initialize $D = ARRAY[n][m]$
5.	For ($i = 1$ to n)
6.	For ($j = 1$ to m)
7.	If ($i = 1$ and $j \neq 1$)
8.	$min = D_{i,j-1}$
9.	Else if ($i \neq 1$ and $j = 1$)
10.	$min = D_{i-1,j}$
11.	Else
12.	$min = MIN(D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1})$
13.	End if
14.	$D_{i,j} = min + a_i - b_j ^p$
15.	End for
16.	End for
17.	Return $dist = \sqrt[p]{D_{n,m}}$

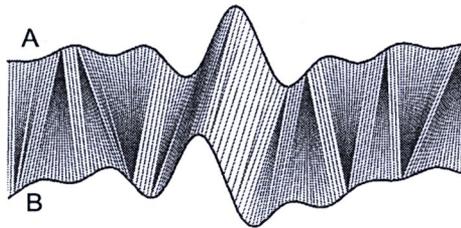


Figure 3.2: Alignment obtained from a DTW distance calculation.

3.1.2 Dynamic Time Warping (DTW) Averaging

DTW averaging was first introduced by Gupta et al. (Gupta et al., 1996) to find an averaged result between two time series sequences. Unlike DTW distance, DTW averaging uses another matrix to store an index of the minimum distance among adjacent elements. The path matrix is created to store an index of the adjacent element that has minimum cumulative distance, and a path is traced back from the last element to the first element. An averaged result is then calculated along the path. Suppose the path $W = \langle w_1, w_2, \dots, w_k, \dots, w_N \rangle$ of length N , where w_k is k^{th} coordinate (i_k, j_k) in the optimal path of sequences $A = \langle a_1, a_2, \dots, a_i, \dots, a_n \rangle$ and $B = \langle b_1, b_2, \dots, b_j, \dots, b_m \rangle$, where i_k and j_k are indices of data points in sequences A and B , respectively. Therefore, a new sequence $Z = \langle z_1, z_2, \dots, z_k, \dots, z_N \rangle$ is derived from elements $z_k = \frac{a_{i_k} \cdot \omega_A + b_{j_k} \cdot \omega_B}{\omega_A + \omega_B}$, where ω_A and ω_B are the weights of sequences A and B , respectively. We also provide a concrete pseudo code of DTW averaging in Table 3.2. For example in Figure 3.3, two sequences $A = \langle 2, 3, 8, 2, 3, 1, 3 \rangle$ and $B = \langle 3, 1, 2, 3, 8, 3, 2 \rangle$ are averaged by DTW averaging algorithm to produce an averaged result $Z = \langle 2.5, 1.5, 2, 3, 8, 2.5, 3, 1.5, 2.5 \rangle$.

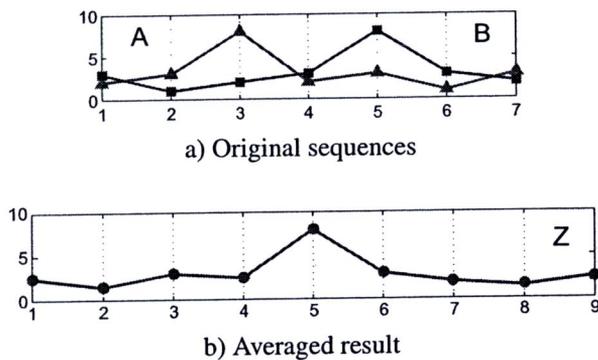


Figure 3.3: Result generated from DTW Averaging

Table 3.2: Pseudo code of Dynamic Time Warping averaging function

FUNCTION $[W] = \text{DTW-AVERAGING} [A, B, \omega_A, \omega_B]$	
1.	$W = \text{WARPINGPATH}(A, B)$
2.	Let N be a length of the path W
3.	Let Z be a time series sequence of length N
4.	For ($k = 1$ to N)
5.	$[i, j] = w_k$
6.	$z_k = \frac{a_i \cdot \omega_A + b_j \cdot \omega_B}{\omega_A + \omega_B}$
7.	Add z_k to Z
8.	End for
9.	Return Z

It is important to note that DTW averaging function is an operation which has only commutative property with no associative property (Niennattrakul and Ratanamahatana, 2007a). In the other words, if there are three sequences A , B , and C , a result of averaging A and B , then C is not necessarily equal to a result of averaging B and C , then A . A sequence ordering can largely affect the averaged result. In addition, an averaging sequence will always be longer or equal to the original sequences. If a large dataset is to be averaged, averaging sequences will be very long which will definitely decrease a system performance. Therefore, in this chapter, two new shape-based averaging functions to resolve this problem and a new averaging scheme to efficiently order averaging sequences are proposed.

3.2 Related Work

Over a decade ago, Gupta et al. proposed a heuristic shape-averaging scheme called NLAAF (Gupta et al., 1996), which was first introduced in signal processing community, and later has been utilized in data mining tasks (Ratanamahatana and Keogh, 2005a; Salvador and Chan, 2007). Specifically, NLAAF uses DTW averaging to produce a mean between a pair of time series sequences. NLAAF consists of two averaging schemes, i.e., NLAAF₁ and NLAAF₂. NLAAF₁ averages sequences in hierarchical manner. Suppose there are eight sequences, i.e., A_1

Table 3.3: Pseudo code of generating a warping path

FUNCTION $[W] = \text{WARPINGPATH}[A, B]$

1. Initialize distance matrix DM and path matrix PM
2. For each a_i in A and b_j in B
3. $DM[i, j] = |a_i - b_j|^p$
4. If ($i = 1$ and $j \neq 1$)
5. $DM[i, j] += DM[i, j - 1]$
6. $PM[i, j] = 1$
7. Else if ($i \neq 1$ and $j = 1$)
8. $DM[i, j] += DM[i - 1, j]$
9. $PM[i, j] = 2$
10. Else if ($i \neq 1$ and $j \neq 1$)
11. $dist = \text{MIN}(DM[i, j - 1], DM[i - 1, j], DM[i - 1, j - 1])$
12. If ($dist = DM[i, j - 1]$)
13. $PM[i, j] = 1$
14. Else if ($dist = DM[i - 1, j]$)
15. $PM[i, j] = 2$
16. Else
17. $PM[i, j] = 3$
18. End if
19. $DM[i, j] += dist$
20. Else
21. $PM[i, j] = 3$
22. Endif
23. Endfor
24. Let n be a length of the sequence X
25. Let m be a length of the sequence Y
26. While ($n \neq 0$ and $m \neq 0$)
27. $w_k = [n, m]$
28. If ($PM[n, m] = 1$)
29. $m = m - 1$
30. Else if ($PM[n, m] = 2$)
31. $n = n - 1$
32. Else
33. $m = m - 1; n = n - 1$
34. End if
35. End while
36. $W = \text{Reverse order of } W$
37. Return W

to A_8 . A_1 and A_2 are averaged to produce $A_{1,2}$, and A_3 and A_4 are averaged to produce $A_{3,4}$, and so on. Then, in the next level, $A_{1,2}$ and $A_{3,4}$ are averaged to produce $A_{(1,2),(3,4)}$, and so on. Limitation of NLAAF_1 is that it requires that the number of sequences must be a power of two. Unlike NLAAF_1 , NLAAF_2 averages sequences in sequential manner. A_1 and A_2 are first averaged to produce $A_{1,2}$, and then $A_{1,2}$ and A_3 are averaged to produce $A_{(1,2),3}$, and so on.

Since NLAAF_1 has limitation that it requires the number of sequences to be a power of two, Gupta et al. recommend to use combination of both NLAAF_1 and NLAAF_2 . For example, to average 100 sequences, 4 sequences will be discarded, and the rest of the sequences will be separated into three groups of 32 sequences, each of which will be averaged using NLAAF_1 . Therefore,

three averaged sequences produced from $NLAAF_1$ will then be averaged using $NLAAF_2$. Since DTW averaging function does not have an associative property, different orderings of sequences in both $NLAAF_1$ and $NLAAF_2$ will lead to different averaged results. Additionally, an averaged sequence from $NLAAF$ will be very long since DTW averaging function will always produce a longer or equal sequence to its original sequences. In this chapter, two new DTW averaging functions and an averaging scheme which produce a more accurate averaged result are proposed, and when this result is used in subsequence clustering, it produces more meaningful clustering results.

3.3 Shape-based Averaging

To average a set of sequences, an averaging scheme to construct an averaged result is proposed since the shape-based averaging does not have an associative property (Niennattrakul and Ratanamahatana, 2007a). Instead of averaging sequences in a random order as done in $NLAAF$, a heuristic solution is introduced to return a good averaged result by averaging a pair of sequences which are the most similar first. After the averaged result is generated, a pair of sequences from the remaining data including the previous averaged result is determined for the next iteration. The scheme keeps going until only one sequence remains. A pseudo code of the averaging scheme is provided in Table 3.4.

Table 3.4: Pseudo code of Shape-based Averaging scheme

FUNCTION $[W] = \text{AVERAGINGSCHEME } [S]$	
1.	Initialize a weight $\omega = 1$ for each sequence S in \mathbb{S}
2.	While ($\text{SIZE}(\mathbb{S}) > 1$)
3.	$[A, B] = \text{Most similar sequences in } \mathbb{S}$
4.	$Z = \text{AVERAGINGFUNCTION}(A, B, \omega_A, \omega_B)$
5.	Remove A and B from \mathbb{S}
6.	$\omega_Z = \omega_A + \omega_B$
7.	Add Z to \mathbb{S}
8.	End while
9.	Return Z

In this chapter, two novel averaging functions, i.e., Cubic-Spline Dynamic Time Warping (CDTW) and Iterative Cubic-Spline Dynamic Time Warping (ICDTW) are introduced. Either one of these two averaging functions can be used as the AVERAGINGFUNCTION in Line 4 of Table 3.4.

3.3.1 Cubic-Spline Dynamic Time Warping (CDTW) Averaging

CDTW averaging function produces a more accurate averaged result by considering both position and amplitude of each data point of a new averaged sequence, while DTW averaging function (Table 3.1) considers only amplitude. In other words, DTW averaging function equally

treats every new data point in a new sequence, while CDTW averaging function additionally determines where a new data point should be placed. Specifically, a position and an amplitude of a data point in the sequence can be observed as x - and y - coordinate in time series. Therefore, the sequence generated from CDTW function is more useful since it preserves both position and amplitude from the warping path. Figure 3.4 shows the comparison between averaged results generated from CDTW and DTW averaging functions, where two inputs are $A = \langle 2, 3, 8, 2, 3, 1, 3 \rangle$ and $B = \langle 3, 1, 2, 3, 8, 3, 2 \rangle$.

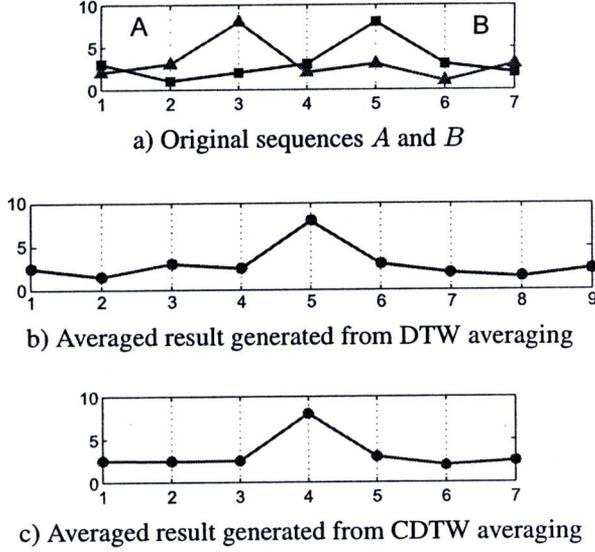
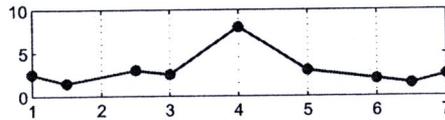


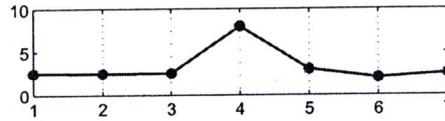
Figure 3.4: Comparison between DTW averaging and CDTW averaging functions

Suppose the path $W = \langle w_1, w_2, \dots, w_k, \dots, w_N \rangle$, where $w_k = (i_k, j_k)$ is k^{th} coordinate in the optimal path of sequences A and B . Therefore, a position z'_{k_x} of a data point in a new sequence Z' is determined by $z'_{k_x} = \frac{\omega_A \cdot i_k + \omega_B \cdot j_k}{\omega_A + \omega_B}$, and an amplitude z'_{k_y} of a data point in a new sequence Z' is determined by $z'_{k_y} = \frac{\omega_A \cdot a_{i_k} + \omega_B \cdot b_{j_k}}{\omega_A + \omega_B}$, where ω_A and ω_B are the weights of sequences A and B , respectively.

However, the length of the sequence Z' is always equal to or longer than two original sequences; therefore, re-sampling is required. In this thesis, CDTW averaging function uses a cubic-spline interpolation (Burden et al., 1997) since it requires no parameter and outperforms other interpolation techniques in re-sampling of natural sequences. Additionally, CDTW function re-samples positions of averaged result to integer values. As illustrated in Figure 3.5, the sequence Z' of 9 data points is re-sampled to the sequence Z of 7 data points. A concrete pseudo code of CDTW function is provided in Table 3.5.



a) Averaged result before re-sampling



b) Averaged result after re-sampling

Figure 3.5: Averaged results before and after re-sampling in CDTW averaging function.

Table 3.5: Pseudo code of Cubic-Spline Dynamic Time Warping (CDTW) averaging function

FUNCTION $[Z] = \text{CDTW-AVERAGING}[A, B, \omega_A, \omega_B]$	
1.	$W = \text{WARPINGPATH}(A, \hat{B})$
2.	Let N' be the length of the path W
3.	Let N be the equal length of time series A and B
4.	Let Z be a time series sequence of size N
5.	Let Z' be a time series sequence of size N'
6.	For ($k = 1$ to N')
7.	$[i, j] = w_k$
8.	$x = \frac{i \cdot \omega_A + j \cdot \omega_B}{\omega_A + \omega_B}$
9.	$y = \frac{a_i \cdot \omega_A + b_j \cdot \omega_B}{\omega_A + \omega_B}$
10.	Add $[x, y]$ to Z'
11.	End for
12.	$Z = \text{CUBICSPLINE}(Z')$
13.	Return Z

3.3.2 Iterative Cubic-Spline Dynamic Time Warping (ICDTW) Averaging

Although CDTW function produces a good averaged result since it considers both position and amplitude, another essential but not necessary condition for averaging is that the averaged result should be in the middle of two original sequences. In other words, DTW distances between the sequences and the result should be equal. Therefore, an iterative approach for CDTW averaging function called Iterative Cubic-Spline Dynamic Time Warping (ICDTW) averaging function is proposed. ICDTW function can truly represent characteristics of a set of subsequences.

It is important to emphasize that the distances between the generated result from CDTW function and two original time series sequences are *not* always equal; therefore, the averaged result needs to be slightly adjusted. Obviously, since all elements in the sequence are real numbers, it is very difficult to obtain the sequence that satisfies this condition; therefore, a heuristic and deterministic solution is proposed, i.e., ICDTW averaging function mentioned above. To average two time series sequences A and B , ICDTW function will find new weights β_A and β_B which make the averaged result Z be the center between the sequences A and B . Obviously, finding both

weights β_A and β_B is not very practical since the weights β_A and β_B are real numbers. A binary search is used instead to find only the weight β_A , when the weight β_B is fixed. Specifically, for each iteration, a new weight β_A is checked whether or not the generated averaged result Z has an equal DTW distances to the sequences A and B . If the distances are equal, ICDTW terminates. In other words, only weight β_A is necessary to search, while weight β_B can be fixed as a constant because two sets of weights are equivalent. For example, for $\{\beta_A, \beta_B\} = \{4, 5\}$, it can be reduced to $\{0.8, 1\}$ when the weight β_B is fixed to 1; therefore, searching for β_A is enough to find any pair of weights $\{\beta_A, \beta_B\}$. Pseudo code of ICDTW averaging function is provided in Table 3.6. Note that two initial weights of A , β_{A_1} and β_{A_2} , are set to be 10^{-5} and 10^5 . These numbers can be initialized to any numbers, where β_{A_1} must be much smaller than β_{A_2} , so the algorithm can be converged.

Table 3.6: Pseudo code of Iterative Cubic-Spline Dynamic Time Warping (ICDTW) averaging function

FUNCTION $[Z] = \text{ICDTW-AVERAGING}[A, B, \omega_A, \omega_B]$	
1.	Initialize weights $\beta_{A_1} = 10^{-5}$, $\beta_{A_2} = 10^5$, and $\beta_B = 1$
2.	Initialize weight $\beta_{A_3} = \frac{(\beta_{A_1} + \beta_{A_2})}{2}$
3.	$Z = \text{CDTW-AVERAGING}(A, B, \beta_{A_3}, \beta_B)$
4.	$d_{Z,A} = \text{DTWDISTANCE}(Z, A) \cdot \omega_A$
5.	$d_{Z,B} = \text{DTWDISTANCE}(Z, B) \cdot \omega_B$
6.	$\beta_{A_3} = d_{Z,A} < d_{Z,B} ? \beta_{A_1} : \beta_{A_2}$
7.	While ($ d_{Z,A} - d_{Z,B} > 0$)
8.	$\beta_{A_3} = \frac{(\beta_{A_1} + \beta_{A_2})}{2}$
9.	$Z = \text{CDTW-AVERAGING}(A, B, \beta_{A_3}, \beta_B)$
10.	$d_{Z,A} = \text{DTWDISTANCE}(Z, A) \cdot \omega_A$
11.	$d_{Z,B} = \text{DTWDISTANCE}(Z, B) \cdot \omega_B$
12.	If ($d_{Z,A} < d_{Z,B}$)
13.	$\beta_{A_2} = \beta_{A_3}$
14.	Else
15.	$\beta_{A_1} = \beta_{A_3}$
16.	End if
17.	End while
18.	Return Z

Note that both CDTW and ICDTW averaging functions can be used in subsequence clustering. However, to preserve characteristics of an averaged result, ICDTW function is more preferred. For CDTW function, the averaged result preserves shape-based averaging process which considers both position and amplitude of the warping alignment, while ICDTW averaging returns more accurate characteristics of the averaged result by calibrating the resulted sequence having the same distance between the result and original sequences. Performance of CDTW and ICDTW functions will be demonstrated in the experiment evaluation.

3.4 Experimental Evaluation

The following experiment will demonstrate the superiority of the proposed averaging functions over the current existing approaches, where the accuracies of the proposed shape-based averaging method, i.e., a new averaging scheme with two proposed CDTW and ICDTW algorithms, comparing with those of NLAAF, are reported. Our proposed methods are evaluated with 20 datasets from the UCR classification/clustering page (Keogh et al., 2011). Table A.1 shows the number of classes, the length of each time series sequence, and the size of the datasets, and Figure A.2 shows some examples of each dataset. Figure 3.6 shows examples of some classes used in this evaluation.

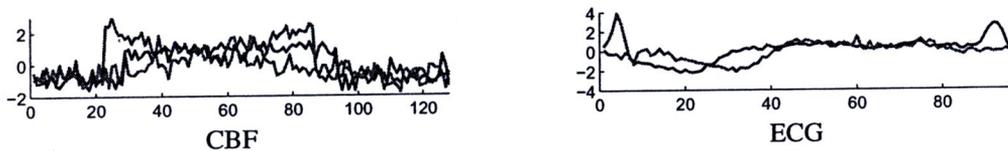


Figure 3.6: Examples of some classes in evaluated datasets.

For each dataset, training data and test data are all combined, and then all sequences are averaged. Note that sequences are averaged within their own classes to achieve maximum utilities. The averaged results are evaluated using SUMDIST function, defined as a summation of all distances between the averaged result and each of the original sequences in the dataset. If a value from SUMDIST is small, it means that this method generates a good averaged result. SUMDIST function is provided as follows.

$$\text{SumDist}(\hat{\mathcal{S}}, \mathcal{S}) = \sum_{i=1}^{|\mathcal{S}|} \text{DTWDistance}(\hat{\mathcal{S}}, \mathcal{S}_i) \quad (3.2)$$

where \mathcal{S} is a dataset, $\hat{\mathcal{S}}$ is the averaged result, and \mathcal{S}_i is each data sequence in the dataset \mathcal{S} .

Table 3.7 shows the SUMDIST comparison between NLAAF and our proposed methods, CDTW and ICDTW functions, where SUMDIST reported in Table 3.7 is a summation of SUMDISTs of all classes. From the experiment results, it is apparent from the experiment results that CDTW and ICDTW functions achieve lower SUMDIST values since all sequences are averaged using a new averaging schemes, while the scheme of NLAAF averages sequences in random manner, and no resampling method is adopted in NLAAF to scale the averaged sequence to the same length. Averaged results from CDTW, ICDTW, and NLAAF of CBF and ECG are shown in Figures 3.7 and 3.8, respectively, where the results from other datasets are provided in Figures

C.1 to C.3 in Appendix C.

Table 3.7: SUMDIST of each averaging method

Dataset	NLAAF	CDTW	ICDTW
50words	4277.6	2348.2	2360.5
Adiac	353.9	285.9	284.3
Beef	384.7	219.9	222.5
CBF	8730.6	4007.1	3821.0
Coffee	69.5	43.0	43.6
ECG	1160.8	528.4	519.4
Face (all)	18339.0	8748.6	8670.4
Face (four)	945.0	613.4	604.4
Fish	516.9	297.3	284.4
Gun-Point	1375.0	466.0	468.4
Lighting-2	2606.9	1195.5	1183.3
Lighting-7	1142.0	858.4	865.1
Oliveoil	6.8	6.5	6.5
OSULeaf	6309.6	2797.4	2805.9
SwedishLeaf	2510.6	1452.8	1415.1
Synthetic	3472.5	2063.1	2050.5
Trace	469.9	221.0	248.5
TwoPatterns	46392.0	1911.4	1874.2
Wafer	635545.3	53026.4	52723.1
Yoga	117113.2	16924.0	16947.9

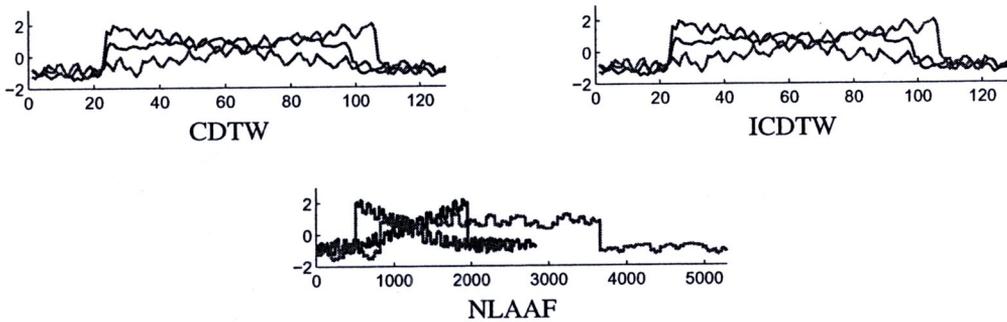


Figure 3.7: Averaged results of CBF

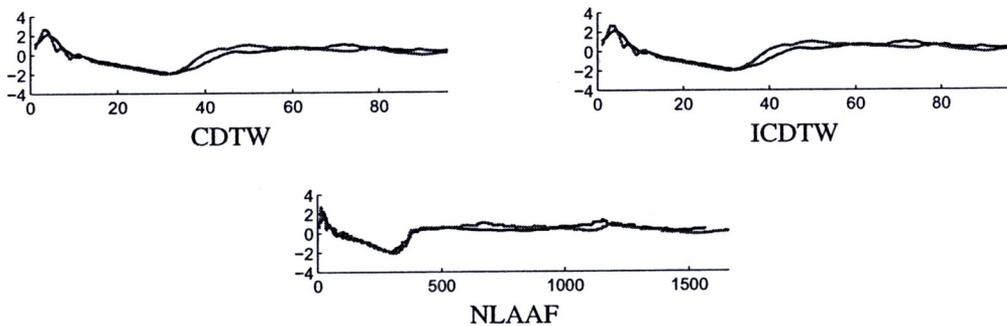


Figure 3.8: Averaged results of ECG

3.5 Averaging Trivial-Matched Subsequences

Trivial-matched subsequences are a set of adjacent subsequences whose differences are only a few points. For example, from a CBF dataset, in Figure 3.9, three sets of trivial-matched subsequences are extracted and shown in Figure 3.9. Therefore, Amplitude Averaging function is inappropriate to average these subsequences since Amplitude Averaging function does not align subsequences before averaging. If Amplitude Averaging is used, the averaged result will be smoothed and the output of subsequence clustering will be meaningless. Figure 3.10 shows the averaged results when Amplitude Averaging averages three sets of trivial-matched subsequences. Since CDTW and ICDTW averaging functions align subsequences before averaging, the averaged result preserves all characteristics, as shown in Figure 3.11 and Figure 3.12, respectively. Therefore, CDTW and ICDTW averaging functions are more appropriate to use to construct cluster representatives in subsequence clustering than Amplitude Averaging function. Either CDTW or ICDTW averaging function can be used to generate cluster representatives in subsequence clustering, where according to experiments, ICDTW provides more accurate averaged results.

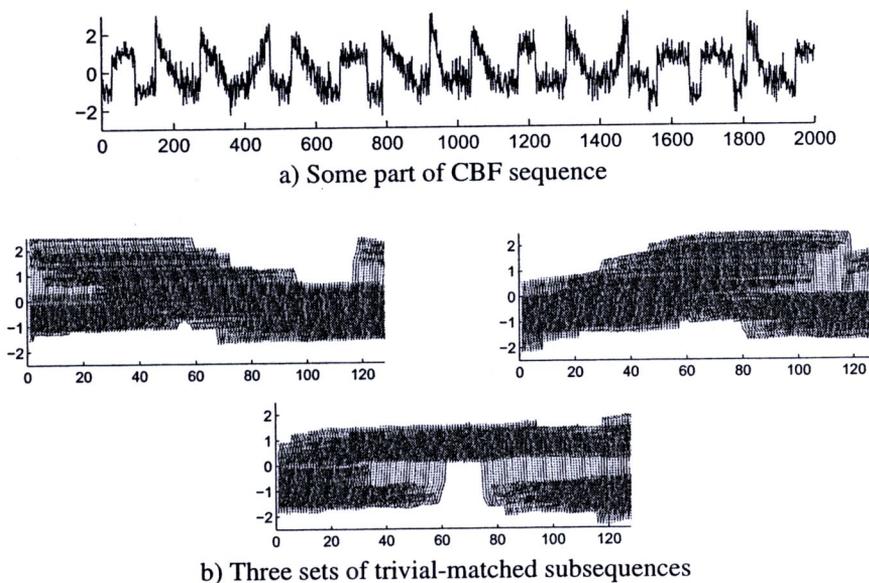


Figure 3.9: Trivial-matched subsequences b) extracted from a) CBF sequence.

3.6 Conclusion

This thesis proposes CDTW and ICDTW functions to generate an accurate averaged result. Since time series data have correlation among dimensions, CDTW and ICDTW functions are more appropriate than Amplitude Averaging function. In addition, CDTW and ICDTW functions are shown to outperform NLAFF, and they should be used as an averaging function for subsequence clustering.

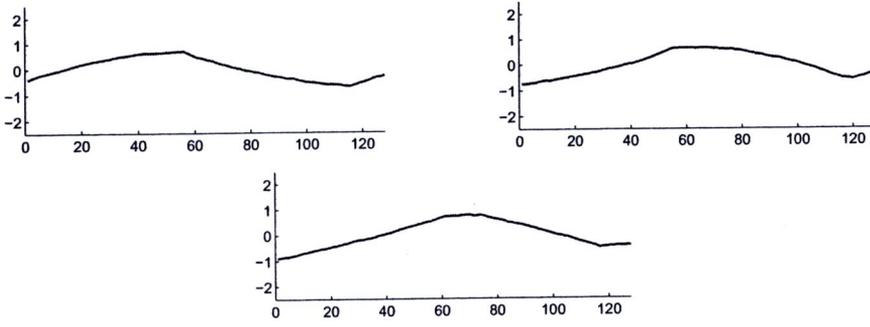


Figure 3.10: Averaged results generated from Amplitude Averaging.

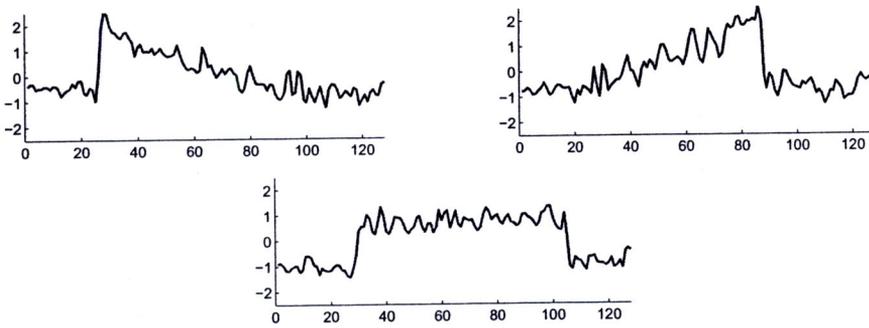


Figure 3.11: Averaged results generated from Shape-based Averaging with CDTW function.

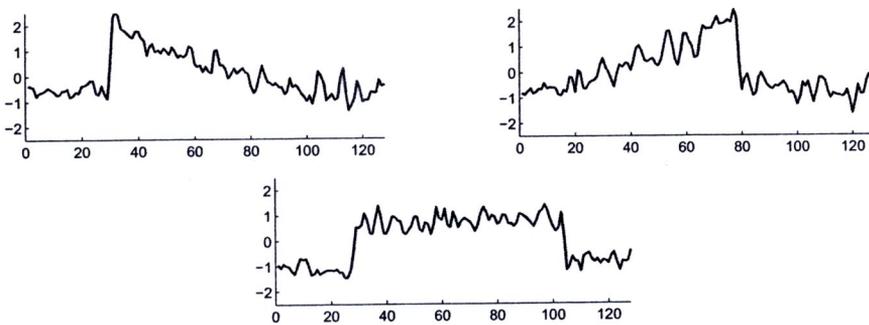


Figure 3.12: Averaged results generated from Shape-based Averaging with ICDTW function.