CHAPTER V

INCREMENTAL SHAPE-BASED AVERAGING

From Chapter III, Shaped-based Averaging is the best solution to construct a representative of a set of subsequences. For streaming applications, a new incoming sequence arrives sequentially in constant time, where an averaged result must be returned for every new incoming sequence. Generally, Shape-based Averaging constructs an averaged result by averaging an entire set of previous sequences. This is obviously impractical for the streaming case, where computational time of constructing an averaged result should not depend on the number of previous sequences which is usually large. Specifically, if there are a lot of previous subsequences, it is not possible to guarantee that a new averaged result will be constructed in time before the next subsequence arrives. Instead of averaging all previous sequences for every new incoming sequence, Iterative Shape-based Averaging creates an averaged result only with a small set of stored sequences. Therefore, time complexity of Incremental Shape-based Averaging depends only on the number of stored subsequences, where the number is much smaller than the number of previous subsequences.

In this chapter, Incremental Shape-based Averaging with two averaging functions, Cubic-Spline Dynamic Time Warping (CDTW) and Iterative Cubic-Spline Dynamic Time Warping (ICDTW) averaging functions, is proposed. The experiments will show that Incremental Shape-based Averaging is much faster than Shape-based Averaging in orders of magnitude, while Incremental Shape-based Averaging maintains low averaging distortion.

5.1 Incremental Shape-based Averaging

Incremental Shape-based Averaging is a method used to incrementally construct averaged result when a set of stored sequences is given with a new incoming sequence. For streaming applications, constructing an averaged result from all previous sequences for every single incoming sequence with limited computational power and storage is simply impractical. Therefore, only some sequences are stored and used to generate an averaged result.

Given a set $\mathbb{T} = \{T_1, T_2, \ldots, T_t\}$ of stored sequences, a set $\mathbb{W} = \{w_1, w_2, \ldots, w_t\}$ of weights of stored sequences, a new incoming sequence S, and the maximum allowance in the number of stored sequences α, where t is a number of stored sequences, Incremental Shape-based Averaging returns an averaged result C. Initially, sets $\mathbb{T}$ and $\mathbb{W}$ are empty, and α is a user-defined

parameter. When a new incoming sequence S arrives, the sets $\mathbb{T}$ and $\mathbb{W}$ are first updated. If the number of stored sequences t is less than the maximum allowance α, S is added to $\mathbb{T}$, and the weight of S, which is initially assigned to 1, is added to the set $\mathbb{W}$; otherwise, a stored sequence T_i which is the most similar to the sequence S under DTW distance is replaced with the averaged result between the sequences T_i and S with weights of w_i and 1, respectively. Therefore, the sets $\mathbb{T}$ and $\mathbb{W}$ are updated with the sequence S, as shown in Table 5.2. After the sets $\mathbb{T}$ and $\mathbb{W}$ are updated, Incremental Shape-based Averaging constructs an averaged result from the copies of $\mathbb{T}$ and $\mathbb{W}$, i.e., $\mathbb{T}_{temp}$ and $\mathbb{W}_{temp}$, by iteratively averaging the most similar pair of sequences within $\mathbb{T}_{temp}$ until only one sequence is left. Its pseudo code is shown in Table 5.3. Note that when the maximum allowance α is positive infinity, to update an averaged result, all previously stored sequences are calculated; therefore, Shape-based Averaging is a special case of Incremental Shape-based Averaging when the maximum allowance $\alpha = \infty$. Pseudo code of Incremental Shape-based Averaging is provided in Table 5.1.

Table 5.1: Pseudo code of Incremental Shape-based Averaging

```
FUNCTION [C] = INCREMENTALSHAPE-BASEDAVERAGING [𝕋, 𝕎, S, α]
1.  [𝕋, 𝕎] = UPDATESTOREDSEQUENCES(𝕋, 𝕎, S)
2.  C = AVERAGESTOREDSEQUENCES(𝕋, 𝕎)
3.  Return C
```

Table 5.2: Updating stored sequences in Incremental Shape-based Averaging

```
FUNCTION [𝕋, 𝕎] = UPDATESTOREDSEQUENCES [𝕋, 𝕎, S, α]
1.  Let t be a number of stored sequences in 𝕋
2.  If (t < α)
3.     Add S in 𝕋
4.     Add w = 1 in 𝕎
5.  Else
6.     dist_Best = INFINITY
7.     For each stored sequence T_i in 𝕋 and w_i in 𝕎
8.        dist = DTW-DISTANCE(T_i, S)
9.        If (dist < dist_Best)
10.          dist_Best = dist
11.          T_Best = T_i
12.          w_Best = w_i
13.       End if
14.    End for
15.    S_avg = AVERAGINGFUNCTION(T_Best, S, w_Best, 1)
16.    Replace T_Best with S_avg
17.    Replace w_Best with w_Best + 1
18. End If
19. Return [𝕋, 𝕎]
```

Table 5.3: Averaging stored sequences in Incremental Shape-based Averaging

FUNCTION $[T_k]$ = AVERAGESTOREDSEQUENCES [$\mathbb{T}$, $\mathbb{W}$]
1. Let $\mathbb{T}_{temp}$ be a copy of $\mathbb{T}$
2. Let $\mathbb{W}_{temp}$ be a copy of $\mathbb{W}$
3. While (SIZE($\mathbb{T}_{temp}$) > 1)
4. $[T_i, T_j]$ = Most similar pair of sequences in $\mathbb{T}_{temp}$
5. T_k = AVERAGINGFUNCTION(T_i, T_j, w_i, w_j)
6. Remove T_i and T_j from $\mathbb{T}_{temp}$
7. Remove w_i and w_j from $\mathbb{W}_{temp}$
8. $w_k = w_i + w_j$
9. Add T_k to $\mathbb{T}_{temp}$
10. Add w_k to $\mathbb{W}_{temp}$
11. End while
12. Return T_k

5.2 Experimental Evaluation

Iterative Shape-based Averaging constructs a new averaged result from only the stored sequences and a new incoming sequence instead of constructing from all previous sequences. Two experiments are designed to demonstrate that Incremental Shape-based Averaging is suitable for streaming applications. The first experiment shows that Incremental Shape-based Averaging is much faster than Shape-based Averaging in orders of magnitude, and the second experiment demonstrates that Incremental Shape-based Averaging, with available storage and computational power, achieves comparable accuracy to Shape-based Averaging with very small distortion, while Incremental Shape-based Averaging is still faster than Shape-based Averaging. Twenty datasets used in this experiment are from the Time Series Clustering/Classification datasets (Keogh et al., 2011). The details of each dataset are provided in Table A.1 in Appendix A, and the examples of each datasets are shown in Figure A.2. In this expeirmental evaluation, two datasets, i.e., CBF and ECG, are mainly used as shown in Figure 5.1.

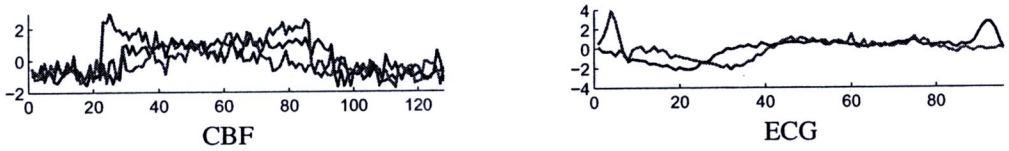

Figure 5.1: Examples of some classes in evaluated datasets.

5.2.1 First Experiment

The first experiment shows the significant speedup of Incremental Shape-based Averaging over Shape-based Averaging, where the maximum allowance α is set to one. For every new incoming sequence, Incremental Shape-based Averaging calculates an averaged result from the

stored sequence, and then this averaged result is used for the next incoming sequence. For Shape-based Averaging, an averaged result is created from all previous sequences for every new incoming sequence which is impractical from streaming data. Figure 5.3 shows time consumption of Incremental Shape-based Averaging compared with Shape-based Averaging using two averaging functions, i.e., CDTW and ICDTW, respectively. From the result, Incremental Shape-based Averaging requires only constant time to update an averaged result, while computational time of Shape-based Averaging grows exponentially. In addition, Incremental Shape-based Averaging is nearly 10^7 times faster. Additional results of this experiment are provided in Appendix E.

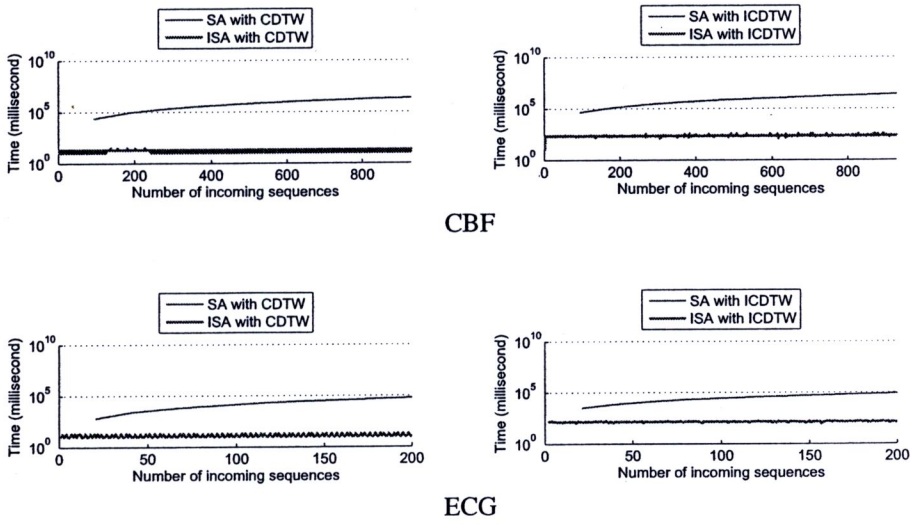

Figure 5.2: Computational time of Incremental Shape-based Averaging and Shape-based Averaging when a new incoming sequence arrives.

5.2.2 Second Experiment

The second experiment shows SUMDIST distance when Incremental Shape-based Averaging is used instead of Shape-based Averaging. Since Shape-based Averaging has no associative property, the updated averaged result from Incremental Shape-based Averaging is not equal to that from averaging all sequences using Shape-based Averaging, where SUMDIST distance can be calculated by the following equation.

$$\text{SumDist}\left(\hat{S}, \mathbb{S}\right) = \sum_{i=1}^{|\mathbb{S}|} \text{DTWDistance}\left(\hat{S}, \mathcal{S}_i\right) \tag{5.1}$$

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

In this experiment, with available computational power and storage, Incremental Shape-

based Averaging can achieve SUMDIST close to Shape-based Averaging, where Shape-based Averaging is a special case of Incremental Shape-based Averaging when the maximum allowance number α is set to a positive infinity. Each class in a dataset is separately evaluated, and SUMDIST of each dataset is reported by summarizing SUMDIST of every class. Difference of SUMDISTs and speedup of Buoy1 and CBF when $k = 3$, $w = 64$, and the maximum allowance number α are varied in percentage to the size of dataset are shown in Figure 5.3 and 5.4, respectively. Figures 5.5 and 5.6 show averaged results generated from Incremental Shape-based Averaging with CDTW and ICDTW, respectively. From the experiment results, Incremental Shape-based Averaging can return averaged results much faster than Shape-based Averaging with only small distortions. Speedup and difference of SUMDIST measured in this experiment is calculated from the time used to update and average sequence of static dataset when the maximum allowance number is varied.

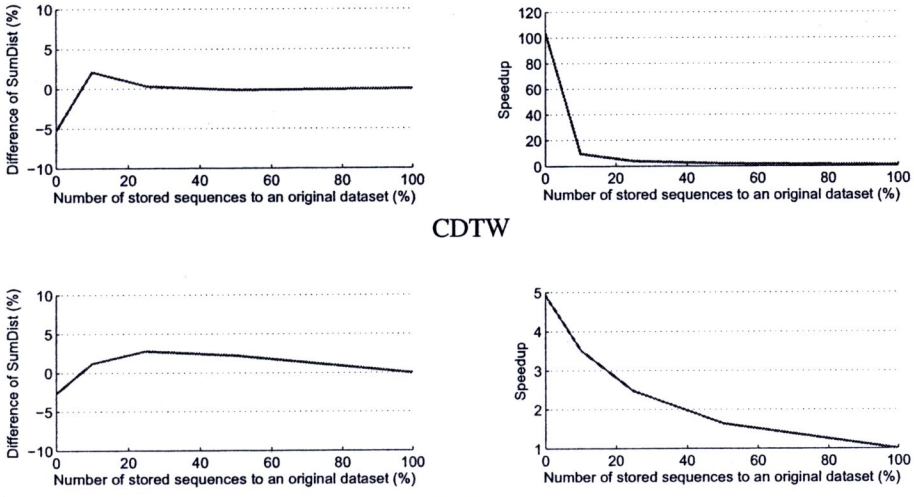

Figure 5.3: Difference of SUMDIST and speedup of Buoy1 when the number of stored sequences to an original dataset is varied.

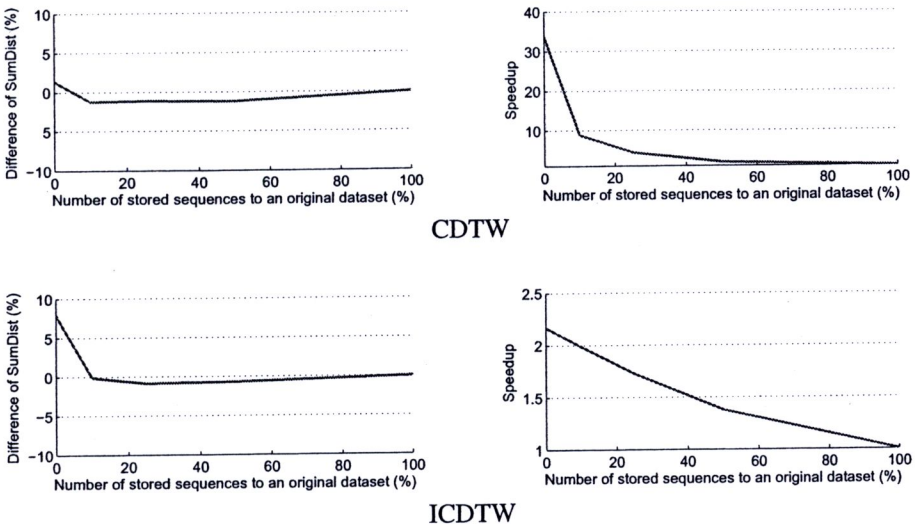

Figure 5.4: Difference of SUMDIST and speedup of CBF when the number of stored sequences to an original dataset is varied.

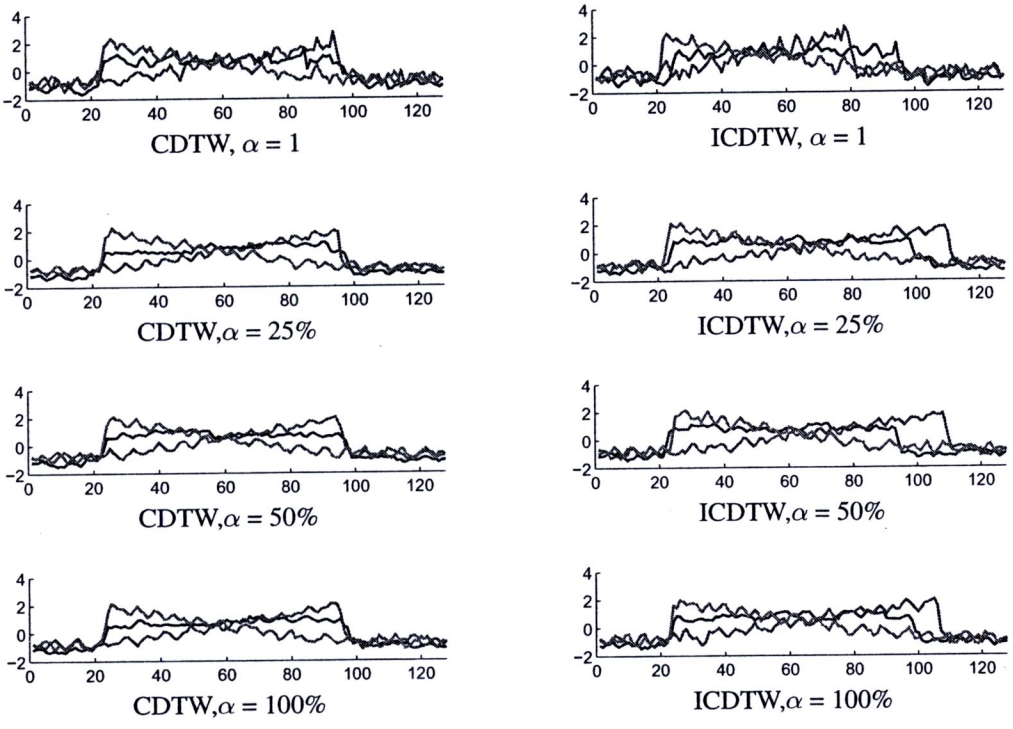

Figure 5.5: Averaged results of some classes of CBF from Incremental Shape-based Averaging.

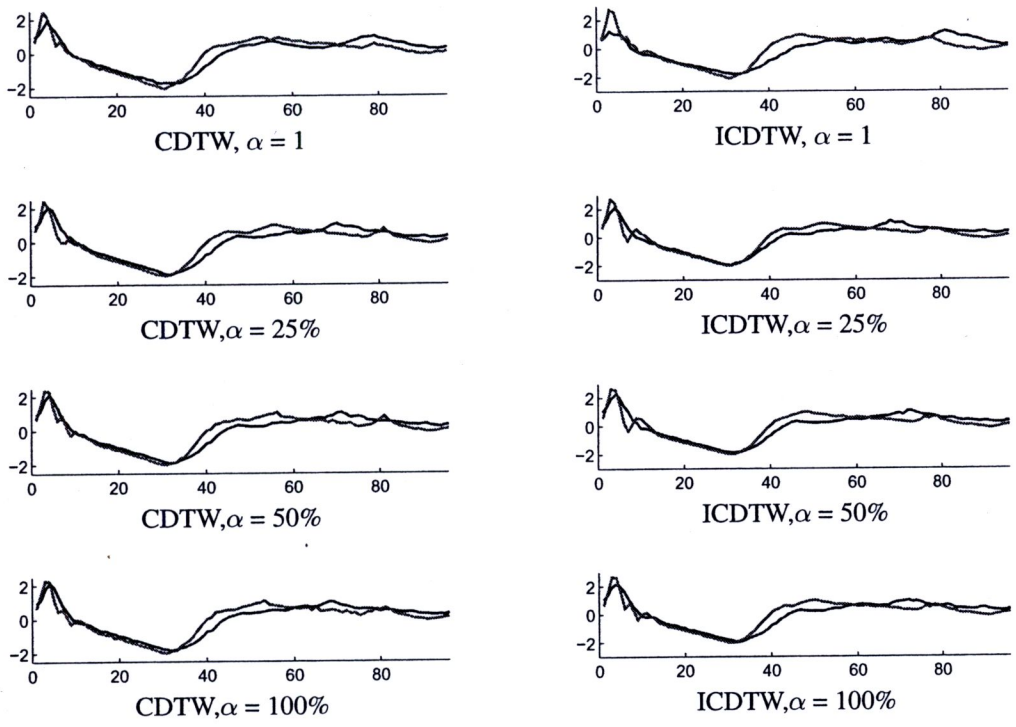

Figure 5.6: Averaged results of some classes of ECG from Incremental Shape-based Averaging.

5.3 Conclusion

Incremental Shape-based Averaging with Cubic-Spline Dynamic Time Warping (CDTW) and Iterative Cubic-Spline Dynamic Time Warping (ICDTW) averaging functions are fast and accurate. To update the averaged result, the stored sequence with its weight is updated to generate a new sequence in constant time. Therefore, instead of constructing an averaged result from all previous data sequences for each and every incoming sequence, Incremental Shape-based Averaging updates only once which reduces computational time in orders of magnitude. In addition, Incremental Shape-based Averaging is proposed to be able to store more than one sequence to increase accuracy if more computational power or storage is available. Moreover, Incremental Shape-based Averaging can be widely extended to construct a shape-based averaged result in streaming applications, whose idea of sequence updates in Shape-based Streaming Subsequence Time Series Clustering (3STSC) is explained in Chapter VI.