An Exploration of DTW with Confidence Intervals

Patrick Liu, Hamza Jamal, Jeremy Kim, Javier Perez

***Abstract* — This paper addresses methodologies to measure the reliability of alignment intervals within the commonly applied framework of Dynamic Time Warping (DTW) [1]. The paper, therefore, focuses on researching novel metrics and visualizations for the evaluation of the alignment confidence across the DTW path, realizing the necessity of digging deeper into specific areas where two signals match or not. Through computational experiments, we introduced "Valley Width" in the cost matrix and sliding window comparison to FlexDTW [2] algorithm.**

I. Introduction

Dynamic Time Warping (DTW) is a technique that aligns two segments of time series data, which may differ in speed across their duration. This algorithm involves computing a pairwise cost matrix and using this to fill in a Dynamic Programming (DP) table. Finding the optimal alignment for DTW is the same problem as finding the lowest-cost path through the DP table, so the maximum cost in this matrix describes the overall quality of the alignment. However, by only considering the final cumulative score, no information is gained surrounding specific confidence intervals along the path. The purpose of this paper is to describe steps taken to characterize the quality of the DTW output at specific points along the signal path rather than just the end.

II. Experimental Setup

For the context of this paper, the sequences of data we used were the chroma features of audio recordings of musical pieces. The team was tasked with creating four scenarios of audio recordings to test our confidence intervals with, which are discussed below and shown in Figure 1.

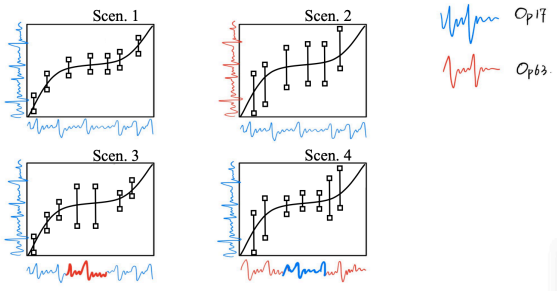

Figure 1. Visualization of four scenarios tested.

The team was also tasked with testing the full alignment case (full recording compared to full recording) and the subsequence alignment case (full recording compared to short query of a recording) for all four scenarios. The team chose to first do full alignment on a possible idea to visualize the confidence intervals and if they worked as expected, we would then test it on the subsequence alignment recordings. Since the work on the full alignment case has not been brought to completion, tests on the subsequence alignment cases have not been conducted.

A. *Scenario 1*

For the first scenario we would run DTW on two different recordings of the same musical piece, Op.17 No. 4. For this scenario we would expect to see high confidence intervals throughout the whole optimal path.

B. *Scenario 2*

For the second scenario we would run DTW on two different recordings of two different musical pieces, Op.17 No. 4 and Op. 63 No. 3. For this scenario we would expect to see low confidence intervals throughout the entire optimal path.

C. *Scenario 3*

For the third scenario we would run DTW on two different recordings of the same musical piece, Op.17 No. 4, except the middle 30 seconds of one of the recordings is replaced with a section of Op. 63 No. 3. For this scenario we would expect to see high confidence intervals throughout the optimal path, except for the middle misplaced section.

D. *Scenario 4*

For the fourth scenario we would run DTW on two different recordings of two different musical pieces, Op. 17 No. 4 and Op. 63 No. 3, where the middle 30 seconds of Op. 63 No. 3 is replaced with a section of Op. 17 No. 4. For this scenario we would expect to see low confidence intervals throughout the optimal path, except for the middle matching section.

III. Introductory Visualization

As we are going to proceed further to introduce the new approaches for the measure of the reliability of DTW alignment, to provide a more detailed and proper background for our new method first, we started from basic analysis about the cumulative cost matrix D and the backtrace matrix B. D keeps the accumulated cost of the alignments; hence, it is used for finding the optimal path. The matrix B stores the predecessor of every cell, for the purpose of construction of a path through backtracking the minimum cost moves.

The visualizations for these matrices are shown in Figure 2. In the D matrix, an optimal path is characterized by a red line that sweeps across a cost gradient from lower, indicative of a well-aligned signal pair. However, scenario 2 clearly indicates increasing time-sharing costs, which match very smoothly, as would be expected with two mismatched signals.

However, Scenarios 3 and 4 are more subtle in nature: even though Scenario 3 does have a reduction in the areas of lower cost after a central mismatch, it becomes quite hard to point out where, exactly, and by how much the mismatch has occurred in the matrix. In such a case, the centers of Scenario 4 coincide with those of Scenario 2. The visual pattern repeats, most likely because the insensitivity effect of the cost gradient to small fluctuations is larger in magnitude than the scale of the maximum and minimum value differences.

The B matrix visualizations, on the other hand, yield limited insights. Across all scenarios, these matrices appear identical and do not reflect the unique characteristics of each scenario's alignment, as shown on the right side of Figure 2. This uniformity underscores the limitations of solely relying on visual inspections of the D and B matrices for detailed alignment analysis

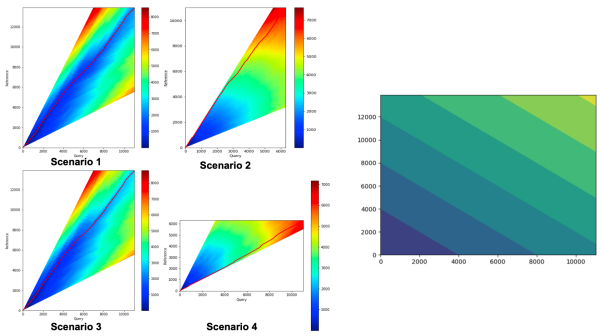

Figure 2. Visualization of the D matrix for all four scenarios (left) and B matrix visualization for all scenarios (right)

IV. Potential Ideas for Confidence Interval Construction

The team explored many ideas trying to find the best one, and had some that we chose not to move forward with. Below are a few we've chosen to speak about in length. Other rejected ideas are scrambling the columns as it negatively affected matching scenarios rather than non-matching and simple backtrace offset as the plots matched for all four scenarios. Note that we did not fully implement the more complex backtrace with offset and can not speak on it's validity.

A. Assessment with Backtrace Voting

To evaluate the robustness of DTW alignment, this study introduces a backtrace voting mechanism. This method aggregates alignment paths by leveraging the comprehensive directional data from the cost matrix and the backtrace matrix. Specifically, the process involves selecting points at regular 50-unit intervals across the cost matrix, which initiates backtrace operations from each point to the origin, and incrementally compiling a vote for each matrix cell traversed by these paths.

The observed patterns varied significantly across scenarios:

a. Scenario 1 demonstrated a mainstream pattern with tributaries, indicative of strong and consistent alignment across the entire audio segment;
b. Scenario 2 revealed a pattern of divergence from the origin, suggesting weaker alignment and potential discrepancies in the audio match;
c. Scenario 3 had the same pattern as Scenario 1;
d. Scenario 4 had a mirrored pattern as Scenario 2;

Interestingly, in Scenarios 3 and 4, we can hardly see any differences in alignments being introduced from Scenarios 1 and 2 with the addition of either non-matching or matching sound clips. This phenomenon further confirms the sensitivity of DTW to the initial positions of the sequences. There is quite a good similarity between the alignment paths of Scenarios 3 and 4 with those of Scenarios 1 and 2, except that it introduced a 30-second non-matching segment into the mix, and it lies in the middle of the tracks. This serves to confirm that the DTW algorithm smoothes out minor local discrepancies, especially when surrounding pieces of audio are also high in similarity, without really causing much interference with the general pattern of the alignment.

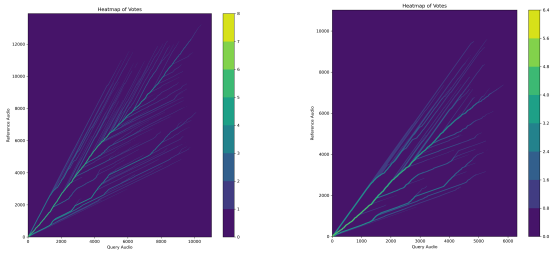

Fig.3 Example of a figure caption. (figure caption)

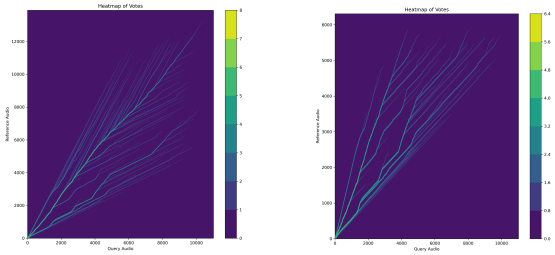

Fig.4 Example of a figure caption. (figure caption)

B. Adding Noise to the Signal

Idea five sought to explore how adding noise at different signal-to-noise (SNR) ratios affects the alignment paths. We predict that when an alignment is strong, minor noise will not significantly alter the alignment path. Conversely, a weak alignment will exhibit substantial changes with the introduction of noise. At each SNR, we added noise to the feature matrices and recalculated the backtrace matrix for

multiple trials, allowing us to directly compare the optimal, ground truth path with the noisy paths.

We applied SNR's of 40dB, 10dB, 5dB, and 2dB with some of the backtrace paths generated shown below. For 40 dB, the recordings are essentially unaffected by noise so we needed to decrease the SNR to see the noise's effect.

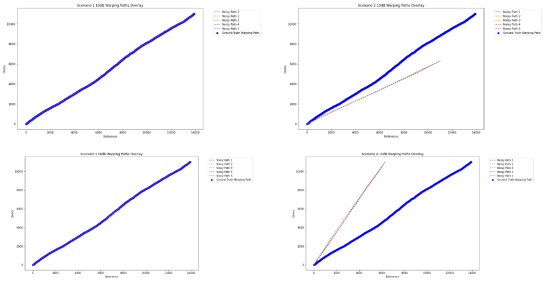

Figure 5. Backtrace paths when SNR = 10dB for each scenario

As Shown in Figure 5, an SNR of 10dB caused Scenario 2 and Scenario 4 to show significant deviation from the optimal path. This result tells us that the backtrace path depends strongly on the initial alignment of the two recordings.

At SNR = 5dB, we have an unexpected result as all four of the scenarios have better alignment than the previous case of SNR=10dB, which is the opposite of what was predicted. As more noise is introduced, the alignment paths should become less accurate.

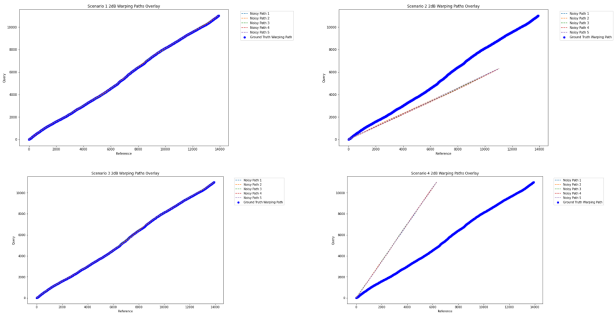

Figure 6. Backtrace paths when SNR = 2dB for each scenario

Finally Figure 6 shows that at SNR=2dB, we see results similar to SNR=10dB, where scenarios two and four explode away from the ground truth path due to their initial mismatching alignment.

In conclusion, based on the seemingly random and unpredictable behavior of the alignment paths (as seen in SNR=10dB, SNR=5dB, and SNR=2dB) we chose to abandon this idea. While the introduced noise at the local level was intended to simulate real-world variability, it also led to increased unpredictability in the alignment paths, making it challenging to interpret the results consistently.

V. Ideas Moving Forward

Below we discuss two ideas we moved forward with and believe have a good chance of giving confidence intervals within DTW.

A. Valley Widths along the Alignment

Our work proposes a novel way to check the sensitivity of starting positions of DTW by quantifying the "valley width" at various points along the optimal path, including the calculation of cost and backtrace matrices used in DTW, followed by an analysis of "valleys" in the cost matrix D. All points on the optimal path were examined for their neighboring values on a line with a slope of -1, to think of these values as troughs where the minimum cost is minimized.

In quantitative terms, the width of valleys was measured by drawing at each point a decision line taken to be that lowest of the average, minimum, and maximum costs along the diagonal and normalizing that metric by the total number of assessed points. This model assumes that broader valleys are in proximity to local misalignments of the optimal path, which have much lower costs than the optimal path and hence mean higher confidence in the alignment. For empirical validation, we initially selected five equidistant points along the optimal path in four different scenarios, calculating the cost of fifty entries along the diagonal for each point. Visualizations from Scenario 1 confirmed the presence of valley-shaped cost distributions along the optimal path (Fig.7).

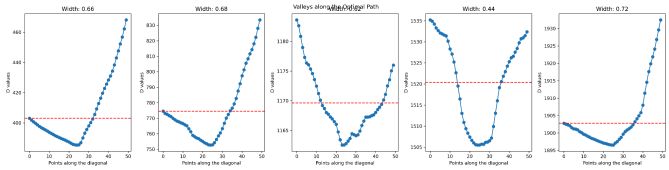

Figure 7. Valleys of points selected on alignment path in Scenario 1

Extending our analysis, we chose 300 equidistant points and broadened our observation interval to 400 points along the diagonal to understand how alignment confidence varies along the path and how inserted audio affects the valley widths. The resulting valley width along the alignment path, filtered by Savitzky–Golay filter, displayed smooth curves with average valley widths of 0.60 and 0.08 for Scenarios 1 and 2, respectively (Fig.8). Notably, in Scenario 3, which included a non-matching audio segment, and Scenario 4, with a matching audio segment from different pieces, distinct patterns were observed — a concave dip in Scenario 3 and a local peak in Scenario 4 occur in the middle. These findings aligned with our hypothesis that a wider valley indicates stronger alignment confidence, allowing us to differentiate between strong and weak alignments along the DTW path.

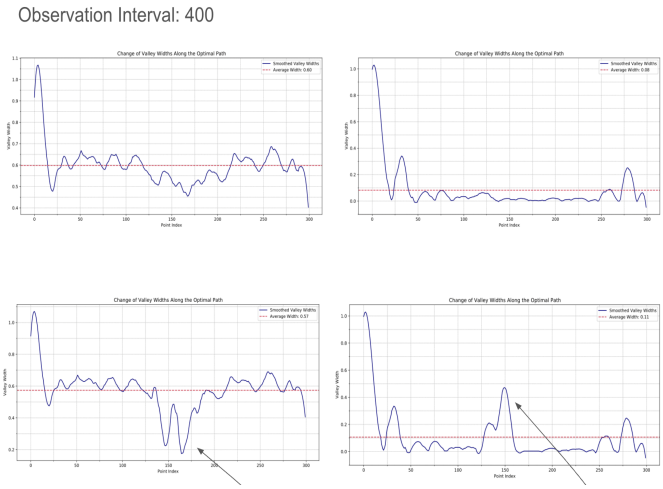

Figure 8. Valley width change along the alignment in 4 scenarios

B. Sliding Window Comparison to FlexDTW

The premise of Idea 7 is that we take small snapshots of the full cost matrix, which we term "windows," and compare the ground truth path that full DTW gives us with output from FlexDTW on that same window. FlexDTW [2], unlike the base DTW algorithm, accounts for many more possible alignment cases by allowing the optimal path to start anywhere on the left or bottom edges of the D matrix, and end anywhere on the top or right edges. Therefore, FlexDTW should only give a similar output to full DTW if the path segment in the analysis window is a good match. Otherwise, the sequences are not truly aligned in this window, and the optimal paths returned by each algorithm will be very different.

In constructing these windows, there are some additional considerations. First, we seek to guarantee that the ground truth path always starts in the bottom left corner and ends on the right edge of each window. To do this, we make the window taller than it is wide by a factor of the most vertical step that DTW can take in its construction of the backtrace matrix. For example, if the steepest step is [3,1], then we make the window 3 times taller than its width.

In doing this, we make sure that each window covers an approximately equal section of ground truth path. Then, by setting a buffer in FlexDTW equal to the width of the analysis window, we force FlexDTW to only consider paths that start at the left edge and end at the right edge. Without this modification, testing revealed that FlexDTW commonly outputs short, "corner" paths, which provide little information about matches in either case.

Once two paths have been generated, we measure their difference as follows: first, we take the average of the highest and lowest x- and y-coordinates among the start points and end points of the paths. In other words, if path 1 goes from (x_{11}, y_{11}) to (x_{12}, y_{12}), and path 2 goes from (x_{21}, y_{21}) to (x_{22}, y_{22}), then the center point is $(\frac{1}{2}\min(x_{11}, x_{21}, x_{12}, x_{22}) + \frac{1}{2}\max(x_{11}, x_{21}, x_{12}, x_{22}), \frac{1}{2}\min(y_{11}, y_{21}, y_{12}, x_{22}) + \frac{1}{2}\max(x_{11}, x_{21}, x_{12}, x_{22}))$. Then we sum the squared distances of each point on each path to this center coordinate, and take the square root. This gives us an RMS path distance, which we use as our error metric.

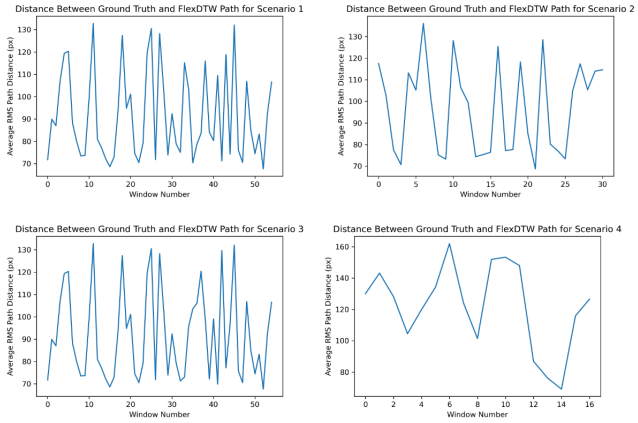

Figure 11. Output of Idea 7 on Scenarios 1, 2, 3, and 4.

The output of this method is depicted in Figure 11. Unresolved bugs prevent us from getting sensible outputs from this algorithm, but what we would expect to see is the inverse of the output for Idea 4: low path distances where the signals match well, and high path distances where the signals match poorly.

VI. Future Work

Based on our current work, the most impactful next steps surround improving the performance of sliding window comparison to FlexDTW algorithm, and converting the currently difficult-to-interpret error metrics of valley width change along the alignment path and difference between FlexDTW alignment & DTW alignment into a more direct statement of the likelihood of a match in each analysis frame. As of now, a 0.68 valley width does not translate to a 68% chance of a match, and an RMS distance of 121 carries little meaning outside of its own limited context. In the interest of time, it was not possible to do so from the beginning, but once these portions of the overall pipeline have been properly tested and refined, we seek to replicate these results in subsequence alignment cases.

VII. Reference

[1] T. Tsai, "Segmental Dtw: A Parallelizable Alternative to Dynamic Time Warping," IEEE Xplore, Jun. 01, 2021.
[2] I. Bükey, J. Zhang, and T. Tsai, "FlexDTW: Dynamic Time Warping With Flexible Boundary Conditions," 2023.