Sliding Adaptive Dynamic Time Warping for Segment Matching in Surgical Simulation Data[⋆]

Alec Cotton[1], Ben Sainsbury[2], and Carlos Rossa[1]

[1] Carleton University, Ottawa ON K1S 5B6, Canada
aleccotton@cmail.carleton.ca
[2] Marion Surgical, Seguin ON P2A 2W8, Canada
ben@marionsurgical.com
[3] Carleton University, Ottawa ON K1S 5B6, Canada
rossa@sce.carleton.ca

Abstract. Virtual reality simulation has become a crucial component in the training of surgeons as it offers a safe and immersive way to develop and improve technical skills while eliminating the need for disposable resources. However, a major limitation of contemporary simulators is that they require access to an expert surgeon during training sessions in order to provide real-time performance evaluation to the trainee. Computer-based methods that provide insightful performance evaluations relative to expert trials have the potential to alleviate the requirement of the presence of an expert during all training sessions. Dynamic time warping (DTW) is a popular algorithm for gesture recognition and evaluation as it can score the similarity between signals as a measure of Euclidean distance. Traditionally, DTW uses a sliding window to identify a given segment (sDTW). However, the segment must be of the same length as the reference segment which is being sought. In gesture recognition from surgical data, this condition is hardly true. This paper proposes a new approach to identifying optimal gesture window sizes using a sliding adaptive dynamic time warping algorithm (saDTW) in which the bounds of an initially fixed window are optimized using simulated annealing in parallel with the initial sliding window. We validate the algorithm in context of gesture recognition in a surgical simulation for percutaneous kidney stone removal. Compared to the tradition sDWT algorithm, the proposed algorithm leads to a 18.45% improvement in identifying a gesture with reference to a reference segment, and a 11.88% increased accuracy in identifying the location of a given gesture embedded in a larger signal.

Keywords: Dynamic Time Warping · Sliding Window · Simulated Annealing.

[⋆] We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) [funding reference number ALLRP 580714 - 22].

Cette recherche a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) [référence ALLRP 580714 - 22]

1 Introduction

Surgical simulation in a virtual environment gives trainees the opportunity to hone their skills in a standardized and safe environment before entering an operating room. While operations in augmented and virtual reality are undoubtedly beneficial in their conservation of resources including surgical instruments and dry and wet models, they only remain beneficial if an expert surgeon is present to guide trainees through each procedure. A systematic method for automatically evaluating and comparing surgical simulation performances remains a challenge [1]. As surgical simulation becomes more prevalent, computer-assisted performance evaluation is in increasing demand. If evaluation metrics could be inferred statistically, the use of simulation tools for surgical procedures could more plausibly be integrated into the training protocol for medical residents, and even practicing surgeons for preparatory measures [2].

Typically, an expert evaluates surgical performance based on specific surgical gestures rather than relying on global performance metrics such as time to completion, average speed, or path length. Therefore, the ideal computer-assisted evaluation algorithm would segment kinematic trajectories into small gestures and provide feedback on short motion segments that are easier to interpret than long surgical tasks [3]. Within the field of gesture segmentation and recognition, there exist two primary subsets of standard approaches for signal processing and recognition: hierarchical measures, including deep learning approaches (RNNs, GRUs, LSTMs, etc.) [3], and non-hierarchical methods, ranging from statistical optimization methods [4][5] to dynamic programming algorithms [6]. The latter approach is often preferred due to limited access to novice and expert surgical simulation data and the large amounts of data required to effectively train deep learning-based algorithms.

One type of non-hierarchical algorithm that can be used to match segments with similar shapes is the dynamic time warping (DTW)[7]. DTW quantifies the similarity between two sequences by constructing a cost matrix of distances between sample points of both sequences. The cost matrix comprises a monotonically increasing path, called the warping path, between the indices of the first and last sample points of each sequence. To find a reference short gesture in a longer new signal, a window containing the reference segment is slid over the new signal until the shortest warping path between them is found [8]. This method, referred to as a sliding window DTW (sDTW) [9], often employs a window of fixed width. Adaptive windowing is explored in [10] to classify multivariate auditory sequences. In [11], human kinetics are inferred from a Kinect Software Development Kit (SDK), after which gestures in 3 degrees-of-freedom (DOF) are identified by manually adjusting the width of the sliding window, i.e., the upper and lower bounds of the cost matrix, to minimize the warping path. In a creative attempt to account for drift in chromatographs as trials deviated from the calibration stage, [12] uses polynomial interpolation to infer the exact warping function applied to the initially calibrated time axis. However, high order polynomials may overfit the signals. Such instances may benefit from spline fitting, however this poses the issue of finding an inverse mapping function with a monotonically increasing time scale. While variants of DTW have been used to compare temporal sequences for segment matching, doing so with a fixed window size limits the algorithm's effectiveness in highly dynamic environments, such as surgical simulation scenarios. Therefore, there is a compelling need to develop an adaptive segment matching algorithm in order to accommodate the complexities inherent to surgical simulation.

This paper introduces the sliding adaptive window dynamic time warping (saDTW) algorithm. Unlike traditional sDTW, the saDTW algorithm employs a variable-length window combined with simulated annealing optimization to adaptively identify matching sequences between two time series. While traditional sDTW can correctly match a window of fixed length to a larger segment, or select one window from a finite list of window lengths employed, the proposed adaptive window with dynamic upper and lower bounds can identify the best matching window of any unspecified length. This ensures that the lowest warping distance in a signal is always found. The proposed saDTW algorithm thus holds the potential to significantly improve the accuracy and efficiency of identifying surgical gestures.

To validate the algorithm, we collected data from a virtual reality surgical simulator for kidney stone removal. A measured gesture S is composed of 6 signals, i.e., the temporal signals representing the position and orientation of the surgical tool. We then define n different nonlinear warping functions, which are applied to S to create n warped versions of S, called S_w. Each newly generated signal S_w is then padded with a synthetic signal generated using Gaussian random walks. We then employ the proposed saDTW algorithm and a traditional sDTW algorithm to attempt to locate S within a larger signal X_w without specifying the warping function, and compare the resulting warping distances. We also employ the saDTW algorithm to predict the time interval over which the gesture S is expected to reside in each X_w. The results show that the proposed algorithm leads to an 11.88% more accurate signal mapping and 18.45% lower warping distances compared to the reference sDTW algorithm.

The remainder of this paper is organized as follows: Section 2 explains the methodology behind the proposed saDTW algorithm. Section 3 introduces the surgical simulator and data collection, followed by experimental validation in Section 4. Section 5 offers a statistical analysis of the saDTW algorithm's performance relative to sDTW, followed by conclusions and future research directions in Section 6.

2 Sliding Adaptive Dynamic Time Warping (saDTW) Algorithm Overview

Let $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_m)$ be two time series. We construct a cost matrix D of Euclidean distances between each point on X and Y as follows:

$$D(i,j) = \|x_i - y_j\| \quad \text{for } i = 1, \ldots, n \text{ and } j = 1, \ldots, m \tag{1}$$

The warping path P^* is a set of continuous matrix elements that defines a mapping between X and Y. The optimal warping path minimizes:

$$P^* = \arg\min_{P^*} \sum_{(i,j)\in P^*} D(i,j), \tag{2}$$

subject to the following constraints:

- P^* starts at $(1,1)$ and ends at (n,m);
- The points in P^* must be monotonically spaced in time;
- Consecutive points in P^* are adjacent or diagonal.

Now consider a signal $X(t)$, and a similar, warped signal $X_w(t)$, where $t = 1, 2, \ldots, n$. We define the segment S as $S = [X(s), X(s+1), \ldots, X(s+k-1)]$ where k is the length of a segment, and s is the starting index, subject to $s \in t$. We consider S to be a segment on $X(t)$ and we seek to identify the corresponding interval on $X_w(t)$ with the lowest warping distance D_w. To this end, we slide a window W of the same length as S across $X_w(t)$. Each window W_i is defined as $W_i = [X_w(i), X_w(i+1), \ldots, X_w(i+k-1)]$, using i as the current index in $X(t)$. The warping distance D_w between S and each W_i is

$$D_w = D(S, W_i) \tag{3}$$

Next, a warping distance threshold T is defined at each window. For each W_i, if $D_w(S, W_i) < T$, a new parallel thread is spawned to initiate the adaptive window algorithm at this location, while the sliding window continues its iterative evaluation on $X_w(t)$.

2.1 Proposed Algorithm

The saDTW algorithm leverages the sDTW algorithm to find a segment S on a warped signal $X_w(t)$. Unlike sDTW, when $D_w < T$ the saDTW algorithm employs simulated annealing on the window bounds in order to identify an index range in which the warping path's cumulative distance is minimized.

In each adaptive window thread, we preprocess the initial lower and upper bounds (ℓ and υ, respectively) of the window for optimization using a set of logical conditions to better approximate a starting point. Let $P_w = \{(S_0, W_{i,0}), \ldots, (S_{k-1}, W_{i,k-1})\}$ be the warping path between the initial iteration of the adaptive window algorithm and the segment S. We count the 0^{th} and $k-1^{th}$ indices of S and W_i in P_w, and adjust ℓ and υ as follows:

$$\ell, \upsilon \leftarrow \begin{cases} \ell + = C_\ell, & \text{if } C_\ell \in W_{i,0} \\ \ell - = C_\ell, & \text{if } C_\ell \in S_0 \\ \upsilon + = C_\upsilon, & \text{if } C_\upsilon \in S_{k-1} \\ \upsilon - = C_\upsilon, & \text{if } C_\upsilon \in W_{i,k-1} \end{cases} \tag{4}$$

where C_ℓ and C_υ count how many times the start and end indices of either signal are reused in P_w. Simulated annealing is employed to find the optimal bounds ℓ and υ for each window W_i in the adaptive stage to minimizes the warping distance D_w between S and W_i [13]. The simulated annealing energy function E to be minimized is the warping distance $D_w(S, W_i)$ given in (3). Initially, ℓ and υ are set to the starting and ending indices of W_i.

The initial lower and upper bound of the sliding window are:

$$\ell_{\text{init}} = W_{i,0}, \quad \upsilon_{\text{init}} = W_{i,k-1}. \tag{5}$$

During each iteration, perturbations $\Delta\ell$ and $\Delta\upsilon$ are randomly generated within a defined neighborhood $\mathcal{N}$, with a zero-centred mean and an adjustable variance hyperparameter σ^2:

$$\Delta\ell, \Delta\upsilon \sim \mathcal{N}(0, \sigma^2) \tag{6}$$

New bounds ℓ' and υ' are then calculated:

$$\ell' = \ell + \Delta\ell, \quad \upsilon' = \upsilon + \Delta\upsilon \tag{7}$$

The energy difference ΔE between the old and new states is:

$$\Delta E = E(\ell', \upsilon') - E(\ell, \upsilon) \tag{8}$$

Starting at a initial temperature $\tau = \tau_0$, the state of the $(k+1)^{th}$ iteration of the simulated annealing algorithm is accepted with probability P, defined as:

$$P_{k+1} = \begin{cases} 1, & \text{if } \Delta E < 0 \\ e^{-\frac{\Delta E}{\tau_k}}, & \text{otherwise} \end{cases} \tag{9}$$

The temperature is updated using a decrement factor $\alpha \in (0, 1)$, i.e.,

$$\tau_{k+1} = \alpha\tau_k \tag{10}$$

The algorithm iterates until τ is less than a predefined threshold $\tau_{\min}$, or until a maximum predetermined number of iterations is reached. The final ℓ and υ are the values that minimize $D_w(S, W_i)$, providing an adaptively optimized window for the dynamic time warping operation.

Parallel threads update a global variable $D_{\min}$ whenever a lower D_w is found:

$$D_{\min} = \min\left[D_{\min}, D_w(S, W_i)\right] \tag{11}$$

Equation (11) discards prospective windows that are no longer contending for the lowest D_w in an identified W_i, keeping computational cost to a minimum. The saDTW algorithm is illustrated in Fig. 1 and the corresponding pseudocode is given in Algorithm 1 below.

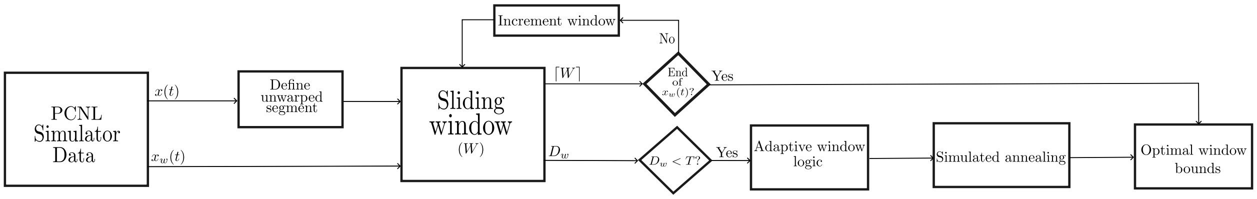

Fig. 1: saDTW data acquisition and handling to optimal window bounds for lowest warping distance relative to the reference segment.

Algorithm 1 saDTW Algorithm Pseudocode

```
Start
Stream PCNL simulator data $X(t)$, $X_w(t)$
$T \leftarrow threshold$
$S \leftarrow [X(s), X(s+k-1)]$
Initialize sliding window $W$ with width $len(segment)$
while $\lceil W \rceil < len(X_w(t))$ do
    Compute $D_w(W, S)$
    if $D_w < T$ then
        Parallelize
        Run adaptive algorithm logic
        Apply simulated annealing to adjust $W$
        Evaluate $D_w$ between adjusted $W$ and reference
    end if
end while
Select the window with the lowest $D_w$ distance as the predicted segment
Display final $D_w$
Stop
```

3 Experimental Setup and Data Collection

PCNL is a procedure to remove kidney stones through an incision on the patient's back when the stones are too large to be passed naturally through the urinary tract. To collect data to validate the proposed algorithm, we use the percutaneous nephrolithotomy (PCNL) surgical simulator from Marion Surgical seen in Fig.(2) [14][15]. The user wears the Oculus Quest 2 goggles to immerse in a virtual reality environment that emulates an operating theatre. The user manipulates a hand-held endoscopic surgical tool in the simulation via a 6-DOF haptic device which records the position and orientation of the surgical tool. Throughout the procedure, the haptic device impedes the operator as they interact with the patient using the endoscopic tool by providing active haptic feedback in order to provide an element of tactile realism during the operation.

In an initial trial, an expert surgeon performs a gesture S while the 6-DOF position and orientation $(X, Y, Z, \alpha, \beta, \gamma)$ of the surgical tool are recorded. We wish to stretch and compress the time axis of S randomly and non-linearly to add variances to the measured samples so that the saDTW algorithm does not simply look for the same sequence of numbers in subsequent time-series data. To that end, referring to Fig. 3, we warp S using the warping function in Algorithm 2: We consider the uniform time steps of S to be S_x, and the magnitude of each sample to be S_y. To warp the time axis non-linearly, we create S_{wx} by sampling $\frac{s+k-1}{r}$ time points from r distinct normal distributions with randomized means (μ) and variances (σ^2), where $s+k-1$ is the number of samples in S. The warping distance between the reference and generated sequences is governed by the hyperparameters r, μ, and σ^2. We then sort the sampled time steps in ascending order. Next, we add Gaussian random noise $\mathcal{N}(\mu, \sigma^2)$ to each sample in S_y, and denote this sequence as S_{wy}.

The result of this process is a sequence of distorted measurements S_{wy} similar to those of S_y, and a warped time axis S_{wx} comprising a random monotonically increasing sequence. When streaming data from the simulator, time steps are taken in the form of indices, and thus a uniform re-sampling must be conducted.

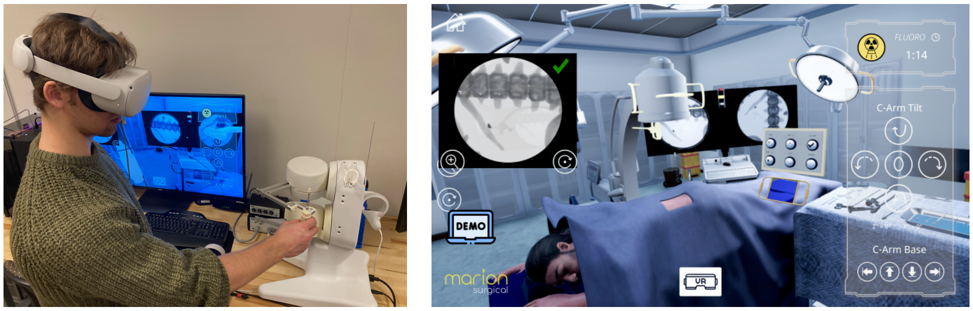

Fig. 2: Marion Surgical's PCNL simulator used for data collection. On the left, the user interacts with the virtual tool via a 6-DOF haptic device. An overview of the virtual operating theatre is shown on the right.

We fit a cubic spline $g(t)$ to the sequences S_{wx} and S_{wy}, and re-sample $g(t)$ on the initial domain of $x_w[t]$ uniformly as seen in the fifth and sixth plots of figure 4.

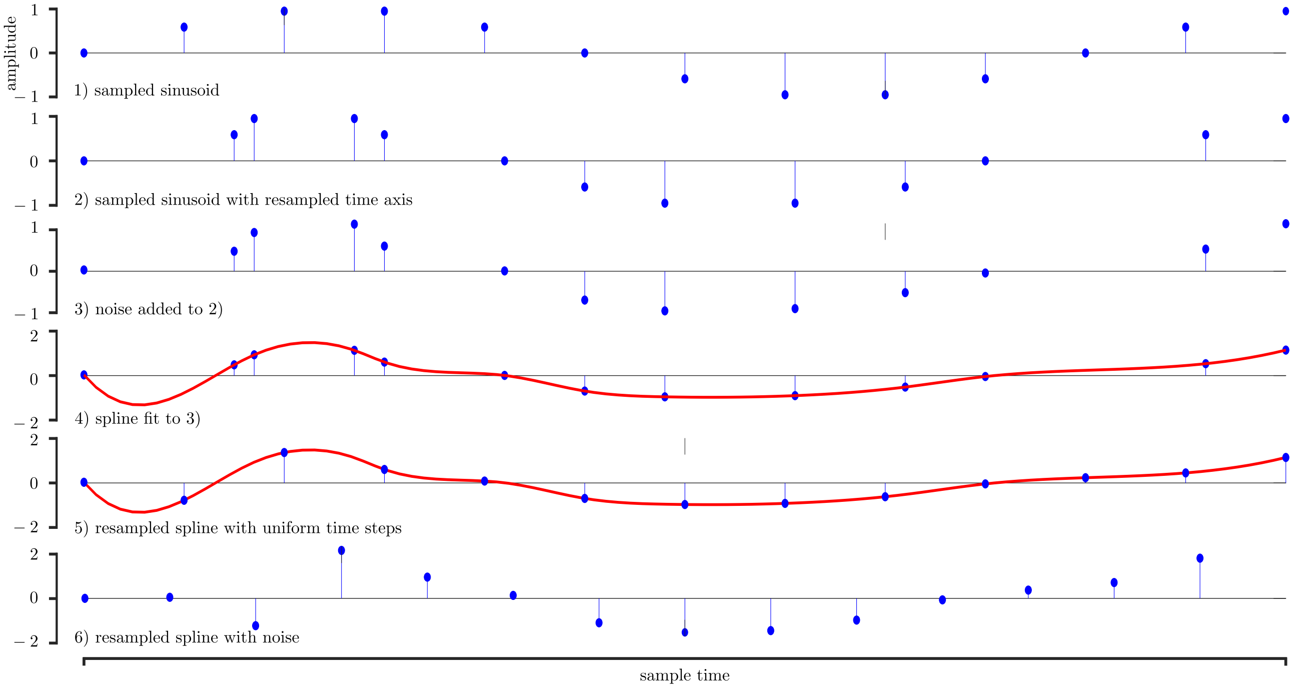

Fig. 3: Synthesized warping of S visualized using an initial uniform sampling of a sinusoid. The time axis is first warped using sampling from r Gaussian distributions, and next noise is added to each discrete sample. A spline is fit to the new warped signal, and it is resampled uniformly to adhere to the integer time step convention that is expected for the proposed algorithm.

The warping transformation function described earlier is called on S 25 times in order to generate a library of warped versions of the segment, denoted as S_{w_i}, which we wish to identify. Each segment S_{w_i} is embedded within a synthesized Gaussian random walk. We refer to the lengthened padded signals from S_{w_i} as X_{w_i}. The location of the embedded gesture S_{w_i} in X_{w_i} is irrelevant, as the sliding window will always begin at the 0^{th} sample of X_{w_i}. Finally, the warped trials in X_{w_i} are ordered such that $D_w(S, S_{w_i}) < D_w(S, S_{w_{i+1}}), \forall i \in \{0, 1, \ldots, 24\}$. We then contrast the saDTW and sDTW algorithms' performances on their ability to identify increasingly warped versions of the initial reference segment within larger signals.

Algorithm 2 Random Signal Warping for DTW Testing

```
Input: Original sequence $S_x = \{0, 1, \ldots, s+k-1\}, S_y = \{y_0, y_1, \ldots, y_{s+k-1}\}$, number of Gaussians $r$, means $\mu = \{\mu_0, \mu_1, \ldots, \mu_r\}$, covariances $\sigma = \{\sigma_0, \sigma_1, \ldots, \sigma_r\}$
Output: Warped sequence $S_{wx}, S_{wy}$
procedure RandomSignalWarp($S_x, S_y, r, \mu, \sigma$)
    $S_{wx} \leftarrow \{\}$
    for $i = 1$ to $r$ do
        $S_{wx} \leftarrow S_{wx} \cup \mathcal{N}(\mu_i, \sigma_i^2, \frac{s+k-1}{r})$
    end for
    Sort $S_{wx}$ in ascending order
    $S_{wy} \leftarrow S_y$
    for $i = 0$ to $s+k-1$ do
        $S_{wy}[i] \leftarrow S_y[i] + \mathcal{N}_i$        ▷ $\mathcal{N}_i \sim \mathcal{N}(\mu, \sigma^2)$ is random noise
    end for
    Fit a B-spline of order 3 to $S_{wx}, S_{wy}$
    $S_{wy} \leftarrow g(N), N \in \mathbb{Z} \cap [0, s+k-1]$
    return $S_{wx}, S_{wy}$
end procedure
```

The proposed saDTW algorithm is validated and compared against the sDTW in two separate ways. In the experimental evaluation, a window containing the original gesture S is slid over each X_w to attempt to locate S with an unknown warping function. As the window slides over X_w, the algorithm calculates the warping distance and when the latter is below a predefined threshold, we spawn a parallel computer thread to adaptively optimize the bounds of the sliding window to further minimize the warping distance. Two metrics are used to evaluate the algorithm's performance:

Warping distance: The warping distance, as defined in (3), gives a degree of confidence in the accuracy of the identified segment, i.e., it quantifies how closely S can be matched to a segment of X;

Interval correctness: The interval correctness compares the predicted with the actual range of a segment in X_w. This is quantified by the Jaccard Index, commonly referred to as the intersection over union (IOU) computed across all DOFs for each S_w. The IOU between each identified range and the true interval is:

$$\mathrm{IoU}(p_0, p_n, t_0, t_m) = \frac{\max(0, \min(p_n, t_m) - \max(p_0, t_0))}{\max(p_n, t_m) - \min(p_0, t_0)} \tag{12}$$

where (p_0, p_n) is the predicted range on S_w from the starting index to the ending index of the adaptive window W, and (t_0, t_m) is the true range of S_{w_i}. The IOU score is on the interval $[0, 1]$, where a perfect identification of the segment of interest would be 1.

4 Experimental Results

The benefit of the adaptive windowing over the fixed window approach can be observed in the example shown in Fig. 4. The top panel shows the reference signal S, whose corresponding S_w is observed in the bottom panel within the index range of 2000-3000 (true range). The sDTW algorithm identifies the window having the smallest warping distance to be between the indices of 1917 to 2917, which results in an IOU equal to 0.847. The reason the algorithms underestimates the true range of the signal is likely due to the fact that S_w has been stretched to cover a larger time interval than the initial sliding window. In contrast, the proposed saDTW algorithm more correctly identifies the signal within the ranges of x to 3000, leading to an IOU of 0.923. The results for each algorithm across all DOFs at tabulated in 1, which displays the average warping distance identified as well as the average IOU score for the saDTW algorithm and sDTW algorithm respectively.

The minimum warping distance recorded for both algorithms for all the 25 S_w used in the validation are shown and compared graphically in Fig. 5. As it can be seen in the figure, the proposed algorithm outperforms the fixed window approach, and, on average, leads to a 18.45% reduction in the warping distance. Next we validate the correctness of the saDTW and sDTW algorithms using the known intervals over which S_{w_i} are

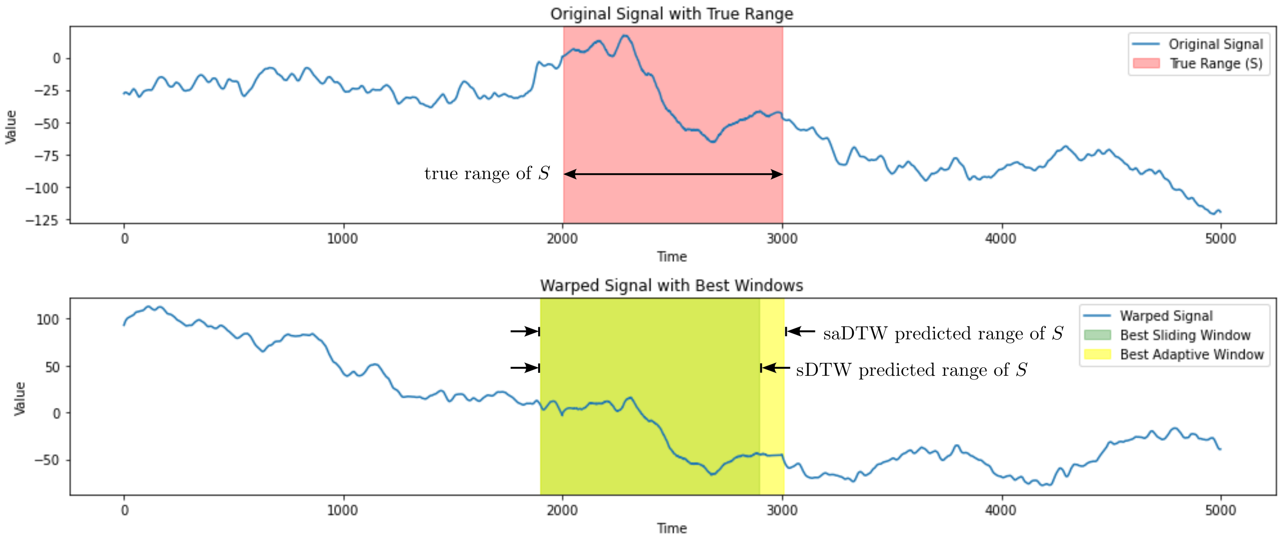

Fig. 4: The convergence of the simulated annealing from the saDTW window is observed to expand upon the boundary of the window with the minimum warping distance on X_w in order to more accurately capture the warped gesture S_{w_i}.

embedded within each larger signal in X_{w_i}. This is done by fitting Gaussians to the IOU results collected across all DOFs for both algorithms for statistical analysis. The comparison of both algorithms' means (μ)and variances (σ) conclude that the saDTW algorithm is 11.88% closer to the ideal identification of each gesture across all DOFs as compared to the sDTW algorithm.

Table 1: Each column of the table corresponds to the saDTW and sDTW algorithms' average warping distance (D_w) and IOU scores across each degree of freedom (translation along X, Y,and Z, as well as rotation about each axis, α, β, γ).

	Average D_w						Average IOU					
Algorithm	X	Y	Z	α	β	γ	X	Y	Z	α	β	γ
sDTW	559.449	236.808	311.944	315.082	323.186	323.279	0.828	0.433	0.102	0.607	0.377	0.399
saDTW	355.436	221.401	306.922	260.931	307.445	288.014	0.763	0.550	0.485	0.724	0.582	0.618

5 Discussion

Figure 6 shows the average and standard deviation of the both algorithms' warping distances across each signal of increasing levels of distortion with S as a reference segment. It is observed that the saDTW algorithm performs better than the regular sDTW method.

The performances of the saDTW and sDTW algorithms in identifying a low warping distance is also computed using a paired T-test:

$$t = \frac{\bar{d}}{\sigma/\sqrt{n}} \tag{13}$$

where $\bar{d}$ is the mean of the differences between the paired observations, σ is the standard deviation of the differences and n is the number of pairs.

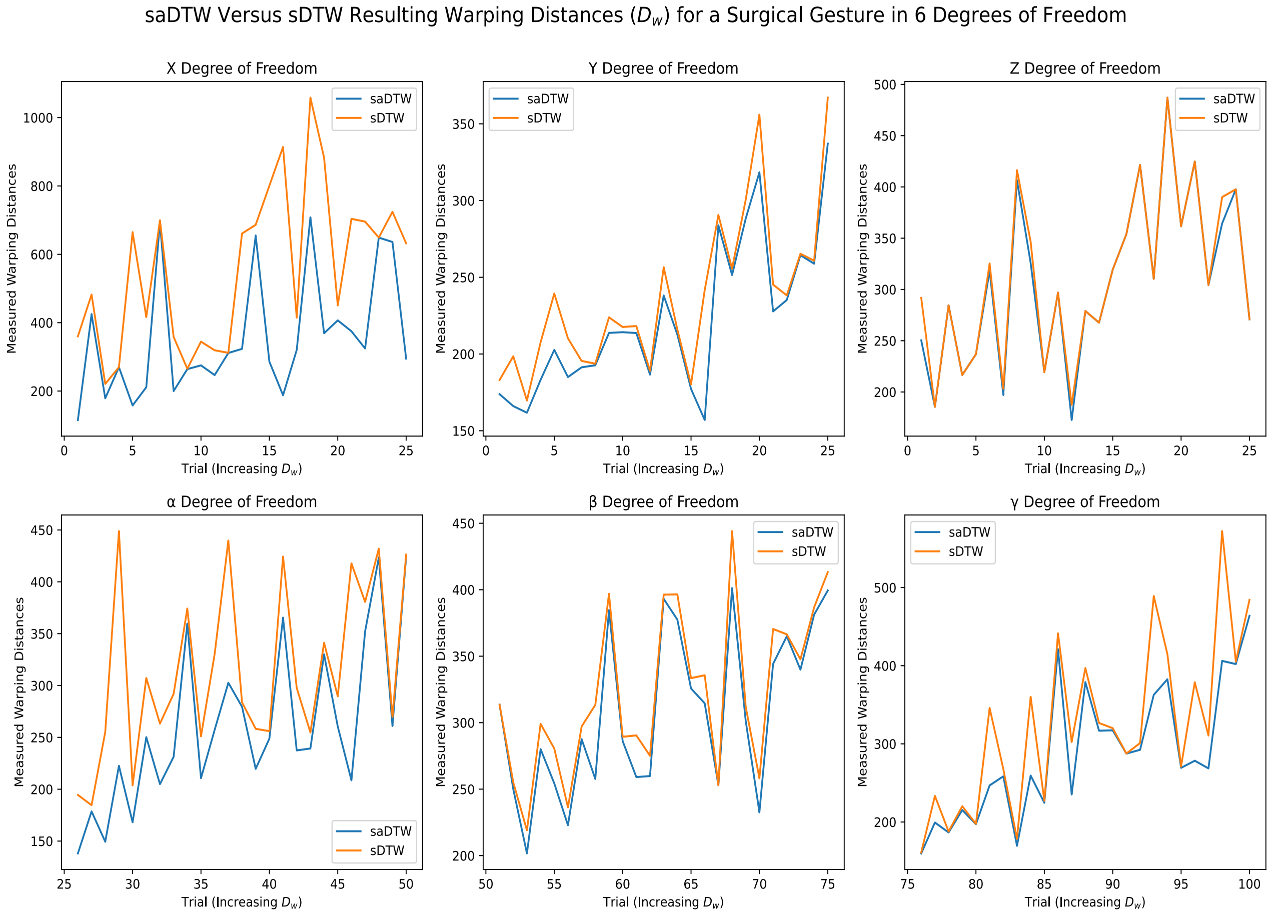

Fig. 5: Minimum warping distances measured by the saDTW and sDTW algorithms when identifying the reference signal S across all trials. X, Y, and Z and α, β, and γ are the translational and rotational DOFs of the surgical tool measured by the simulator.

The Gaussians fit to each algorithm's resulting IOU scores are compared by considering the ideal case wherein gestures have been correctly identified with perfect IOU scores and consistency across all DOFs. In this instance, we would obtain a Gaussian with a mean vector $\mu = \mathbf{1}_{6\times1}$, and a covariance matrix $\Sigma = \mathbf{0}_{6\times6}$. We evaluate the saDTW and sDTW algorithms individually by taking the Euclidean distance of each of their mean vectors to that of the ideal case, and the Frobenius distance between each their covariance matrices and that of the ideal case. We then quantify how much closer one algorithm is to the ideal case using the two aforementioned metrics. The Euclidean distance and Frobenius distance between means and covariances respectively are computed as follows:

$$d(\mu, \mathbf{1_{6\times1}}) = \sqrt{\sum_{i=1}^{6}(\mu_i - 1)^2} \tag{14}$$

$$\|\mathbf{\Sigma} - \mathbf{0}_{6\times6}\|_F = \sqrt{\sum_{i=1}^{6}\sum_{j=1}^{6}\mathbf{\Sigma}_{ij}^2} \tag{15}$$

Where μ is the mean of a Gaussian fit to either the saDTW or sDTW algorithm, and σ is the covariance matrix for the respective Gaussian.

The result is summarized in Fig. 6. The paired T-test yields a T-statistic of -3.767759, and a p-value of 0.000946, strongly indicating that the saDTW algorithm consistently converges to a window with a lower warping distance than that of the sDTW algorithm. In comparing the saDTW and sDTW algorithms' IOU scores by fitting Gaussians to each respectively across all DOFs and using equations 14 and 15 to quantify the degree to which their means and variances adhere to the distribution describing an ideal identification of each gesture, we observe that the saDTW algorithm is 11.88% more effective in identifying the correct interval over which a gesture is expected to be observed than the reference sDTW algorithm. We can thus conclude that the saDTW algorithm significantly improves the convergence of a sliding window to a minimal warping distance compared to the regular sDTW algorithm.

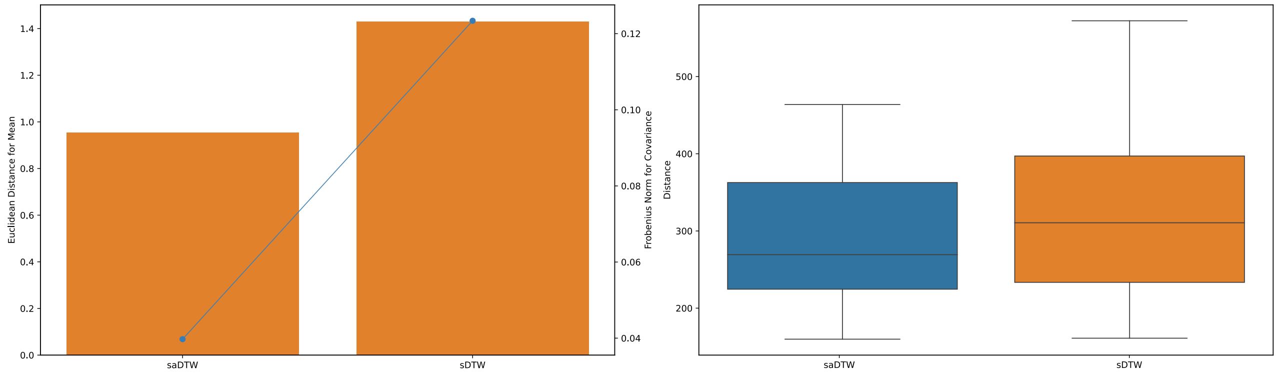

Fig. 6: Comparative performance analysis of saDTW and sDTW algorithms. IOU performance in gesture recognition task (left). Warping distance performance across all trials (right).

6 Conclusion and Future Work

This paper presents a new variation of the DTW algorithm, leveraging the existing sliding window feature to match sequences embedded within larger signals to a reference segment. In addition to identifying a window with a minimal D_w, the saDTW algorithm spawns parallel computing threads at windows below a certain distance threshold to employ a simulated annealing-backed adaptive window optimization, resulting in the identification of a window within which the segment more confidently resides.

In the context of surgical trainee evaluation and gesture detection using a specialized haptic PCNL simulator in virtual reality, the saDTW algorithm poses a statistically significant improvement relative to the reference sDTW algorithm both in its ability to identify segments with a lower warping distance D_w (18.45%), and in its accuracy of gesture recognition, measured by each prediction's IOU with respect to the ground truth intervals on which the gestures were known to be (11.88%).

Future work will focus on the development of real-time gesture recognition algorithms to carry out segment matching tasks as data is streamed lived from the PCNL surgical simulator. In addition to overcoming the real-time challenge inherent to dynamic programming algorithms such as DTW, further analysis of segments from live trials is required in order to provide the user with informative feedback beyond statistical similarity to a reference trial. For example, for the purpose of implementing instantaneous visual trajectory cues, or haptic feedback as guidance.

References

1. P. Dubois, Q. Thommen, and A. Jambon, "In vivo measurement of surgical gestures," *IEEE Transactions on Biomedical Engineering*, vol. 49, no. 1, pp. 49–54, 2002.
2. O. Wilz, B. Sainsbury, and C. Rossa, "Constrained haptic-guided shared control for collaborative human–robot percutaneous nephrolithotomy training," *Mechatronics*, vol. 75, p. 102528, 2021.
3. B. van Amsterdam, M. J. Clarkson, and D. Stoyanov, "Gesture recognition in robotic surgery: a review," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 6, 2021.
4. C. J. Pérez-del Pulgar, J. Smisek, I. Rivas-Blanco, A. Schiele, and V. F. Muñoz, "Using gaussian mixture models for gesture recognition during haptically guided telemanipulation," *Electronics*, vol. 8, no. 7, 2019.
5. J. Rosen, J. Brown, L. Chang, M. Barreca, M. Sinanan, and B. Hannaford, "The bluedragon - a system for measuring the kinematics and dynamics of minimally invasive surgical tools in-vivo," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 2, pp. 1876–1881 vol.2, 2002.
6. F. Despinoy, D. Bouget, G. Forestier, C. Penet, N. Zemiti, P. Poignet, and P. Jannin, "Unsupervised trajectory segmentation for surgical gesture recognition in robotic training," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1280–1291, 2016.
7. J. M. Carmona and J. Climent, "A performance evaluation of hmm and dtw for gesture recognition," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* (L. Alvarez, M. Mejail, L. Gomez, and J. Jacobo, eds.), (Berlin, Heidelberg), pp. 236–243, Springer Berlin Heidelberg, 2012.
8. S. Riofrío, D. Pozo, J. Rosero, and J. Vásquez, "Gesture recognition using dynamic time warping and kinect: A practical approach," in *2017 International Conference on Information Systems and Computer Science (INCISCOS)*, pp. 302–308, 2017.
9. G. Plouffe and A.-M. Cretu, "Static and dynamic hand gesture recognition in depth data using dynamic time warping," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 2, pp. 305–316, 2016.
10. N. Gillian, B. Knapp, and S. O'modhrain, "Recognition of multivariate temporal musical gestures using n-dimensional dynamic time warping.," in *Nime*, pp. 337–342, 2011.
11. H. Hiyadi, F. Ababsa, C. Montagne, E. H. Bouyakhf, and F. Regragui, "Adaptive dynamic time warping for recognition of natural gestures," in *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 1–6, 2016.
12. J. T. Prince and E. M. Marcotte, "Chromatographic alignment of esi-lc-ms proteomics data sets by ordered bijective interpolated warping," *Analytical Chemistry*, vol. 78, no. 17, pp. 6140–6152, 2006. PMID: 16944896.
13. S. Feld, C. Roch, T. Gabor, X.-T. M. To, and C. Linnhoff-Popien, "The dynamic time warping distance measure as q u bo formulation," in *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, pp. 946–950, 2020.
14. B. Sainsbury, M. Łacki, M. Shahait, M. Goldenberg, A. Baghdadi, L. Cavuoto, J. Ren, M. Green, J. Lee, T. D. Averch, and C. Rossa, "Evaluation of a virtual reality percutaneous nephrolithotomy (pcnl) surgical simulator," *Frontiers in Robotics and AI*, vol. 6, 2020.

15. J. R. M. G. M. F. B. Sainsbury, O. Wilz and C. Rossa, “reoperative virtual reality surgical rehearsal of renal access during percutaneous nephrolithotomy: A pilot study,” *MDPI Electronics, section Computer Science Engineering, Advances in Tangible and Embodied Interaction for Virtual and Augmented*, vol. 11, p. 1562, 2022.