Dynamic Speed Warping: Similarity-Based One-shot Learning for Device-free Gesture Signals

Xun Wang, Ke Sun, Ting Zhao, Wei Wang, and Qing Gu
State Key Laboratory for Novel Software Technology, Nanjing University
{xunwang,kesun,tingzhao}@smail.nju.edu.cn, kesun@eng.ucsd.edu, {ww,guq}@nju.edu.cn

***Abstract*—In this paper, we propose a Dynamic Speed Warping (DSW) algorithm to enable one-shot learning for device-free gesture signals performed by different users. The design of DSW is based on the observation that the gesture type is determined by the trajectory of hand components rather than the movement speed. By dynamically scaling the speed distribution and tracking the movement distance along the trajectory, DSW can effectively match gesture signals from different domains that have a ten-fold difference in speeds. Our experimental results show that DSW can achieve a recognition accuracy of 97% for gestures performed by unknown users, while only use one training sample of each gesture type from four training users.**

I. Introduction

Device-free gesture recognition systems use the Radio Frequency (RF) [1]–[9] or sound signals [10]–[15] to detect and recognize human movements. By analyzing the signal reflection of the hand, device-free sensing allows users to interact with their devices freely without wearing any sensor. Such natural and unconstrained interaction paradigm would become a vital component for the next generation Human-Computer Interaction (HCI) solutions.

One of the key challenges for device-free sensing is to robustly recognize gesture signals for different users and in different environments. Traditional machine learning methods use *large datasets* and *intensive training process* to extract domain-independent features from gesture signals. For example, one can collect gesture samples in different domains and use Generative Adversarial Networks (GANs) to reduce the impact of domain-specific features [16], [17]. However, due to the insufficient understanding of the machine-generated models, the performance of these domain-independent models under a new environment cannot be guaranteed. Fine-tuning the model in a new domain may require a large number of new samples to be collected and labeled by the end-user in the new environment. Even if virtual samples can be produced via geometric models using a small number of gestures in the target domain [18], the retraining process still incurs formidable costs for mobile systems.

In this paper, rather than extracting domain-independent feature, we use Dynamic Speed Warping (DSW) to derive a similarity measure between device-free gesture signals. As users may perform the gesture with different speeds and the Doppler shift largely depends on the environment [19], speed variations lead to severe robustness issues in the widely-used speed-based gesture features [1]–[3]. By removing the speed variation, the DSW similarity enables domain-independent one-shot learning that learns information about object categories from one, or only a few, training samples. Thus, it reduces both the data collection and training cost. The design of the DSW algorithm is based on the critical observation that the gesture type is determined by the trajectory of hand components, e.g., fingers and the palm, rather than the movement speed. We show that the similarity in trajectory leads to the similarity in *the total movement distances* and *the scaled speed distributions*. The total movement distances are similar because considering the specific trajectory for a gesture, *e.g.*, click, the starting and the ending postures of the hand remain the same, no matter how fast the user performs the gesture. The scaled speed distributions are similar because when the user changes the movement speed, the speeds of different parts of the hand, such as the fingers and the palm, changes proportionally. Therefore, the speed distribution of different components of a fast gesture movement can be matched to the distribution of a slow gesture movement of the same type, when we scale down speeds of all components by the same factor. Based on this observation, we design a dynamical programming algorithm, which is inspired by Dynamic Time Warping (DTW) [20], to calculate the similarity of gesture signals in terms of the total movement distance and the scaled speed distribution.

The DSW similarity measure leads to new ways to explore the gesture recognition problem. First, the robust gesture matching algorithm can be combined with kNN to serve as a similarity-based one-shot learning scheme that only requires a small number of training samples. As the DSW algorithm can adapt to different gesture speeds, it dramatically reduces the data collection/labeling cost and can incrementally tune the system without retraining. Second, the DSW similarity measure can serve as the basis for unsupervised or semi-supervised learning systems. The DSW algorithm can automatically derive the type of gestures of unlabeled samples by clustering them using the speed-independent measure.

We perform extensive evaluations of DSW using ultrasound-based gesture signals. Our experimental results show that DSW can achieve a recognition accuracy of 97% for gestures performed by unknown users, while using only one training sample of each gesture type from four training users. DSW also outperforms the DTW algorithm in all three external indicators for clustering performance. Therefore, DSW similarity can serve as a powerful tool for both supervised and unsupervised learning tasks.

The main contributions of our work are as follows:

- We propose a new similarity measure that can adapt to the speed variations in gesture signals of different domains.
- We formally prove the properties of the speed adaptive signal matching scheme and show that the result of DSW is a valid similarity measure.
- Using real-world ultrasound gesture signals, we show that the DSW algorithm can serve as a solution for both one-shot learning in supervised gesture recognition and unsupervised gesture clustering tasks.

II. Related Works

Existing works that are closely related to our approach can be categorized into three areas: domain-independent feature extraction, cross-domain adaptation, and DTW-based schemes.

Domain-independent feature extraction. Early device-free gesture recognition systems use statistical values (mean, variance, *etc.*) of the signals [21]–[23] or Doppler speeds [24], [25] as the gesture features. However, it's well known that these features are dependent on the user, the location of devices, and multi-path conditions introduced by the environment. There are two major approaches to extract domain-independent features for device-free gesture signals. The first approach is to use an adversarial network as domain discriminator to help the feature-extracting network in generating domain-independent features [16]. However, the training process requires huge datasets from multiple domains, which leads to high data collection costs. The second approach is to use geometric models to recombine signals measured through multiple links into a domain-independent body-coordinate velocity profile [19]. However, this domain-adaption method uses multiple devices and assumes that accurate user locations are known.

Cross-domain adaptation. Instead of using domain-independent features, we can also transfer a domain-specific gesture recognition model into the target domain. One approach is to use transfer learning schemes to retrain the model using a small number of samples in the target domain [26]–[28]. Another way is to use neural networks or geometric models to transfer the samples in the source domain to the target domain in order to boost the number of training samples in the target domain [18], [29]. Compared to these approaches that need samples in the target domain for bootstrapping, the DSW scheme can evaluate the similarity of gestures from unknown target domains.

Dynamic Time Warping schemes. The DTW algorithm is originally designed for matching speech signals that have different lengths in time [20]. As human activities also have variable durations, DTW has been adopted for various types of activity recognition systems [30], [31]. In device-free gesture recognition, DTW has been applied for matching either the raw gesture signals [32], [33] or the extracted features [21], [34]. However, these DTW applications only consider the scaling in time rather than the scaling of speed distribution and the consistency of movement distance.

III. Device-free Gesture Matching

In this section, we first summarize the state-of-the-art gesture matching methods and their limitations. We then describe our insight on the characteristics of device-free gesture signals. Finally, we demonstrate the benefits of using such speed-adaptive characteristics for gesture similarity calculation.

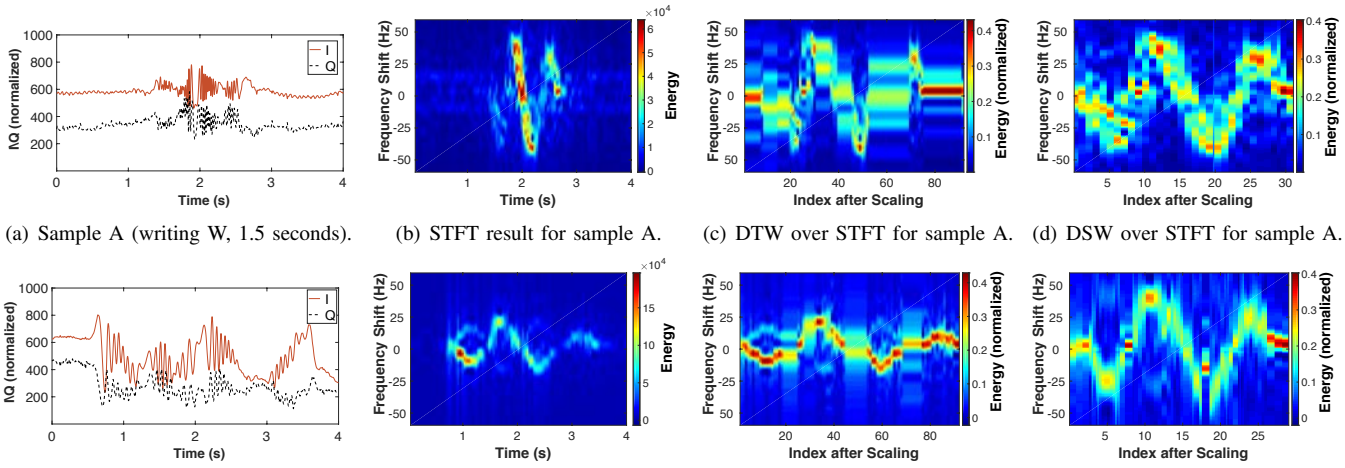
A. Gesture Matching Methods

Device-free gesture recognition systems collect radio/sound signals reflected by the hand to perform gesture recognition. We call these radio/sound signals gesture signals. The most-widely used gesture signals are complex-valued baseband signals that have Doppler frequencies corresponding to the hand movement speeds [1], [3], [25]. For instance, Figure 1(a) and 1(e) show the ultrasonic baseband signals of two samples of the *writing "W"* gesture, where the user writes the letter "W" in-the-air by moving hand back-and-forth twice. The I-component and the Q-component are the real and imaginary parts of the gesture signal. Meanwhile, Figure 1(b) and 1(f) show the corresponding spectrogram calculated through Short-time Fourier transform (STFT). We can observe that the gesture signal has a negative Doppler frequency when the hand is moving away and has a positive frequency when the hand is moving back. Thus, the gesture signals show specific patterns that can be matched with gesture movements and actions.

At early stages of device-free gesture recognition, researchers collect statistical parameters from gesture signals as features, such as mean, variance, and standard deviation. Such statistical features cannot adapt to speed variations and often lead to models that are not robust enough for real-world applications. Some other unsuccessful efforts indicate that linear transformation is inherently insufficient for handling the complicated nonlinearity fluctuation in patterns [25].

The DTW algorithm is a pattern matching algorithm with a nonlinear time-normalization effect. While directly applying DTW on the gesture waveforms have been applied in device-free keystroke matching, it only works when gestures start from a fixed point and is not robust enough for daily gesture recognition tasks [32]. As an example, consider the gesture samples in Figure 1(a) and 1(e), which have durations of 1.5 and 3 seconds, respectively. Besides the differences in frequencies caused by different movement speeds, the two waveforms also have different initial phases and different low-frequency components. The initial phase and low-frequency components of the gesture signal depend on the starting position and small body movements [3], which are noise for gesture recognition. However, these noisy factors dominate the similarity calculated by DTW so that DTW could not find a suitable matching for these two samples in the time domain.

Fortunately, STFT and other time-frequency analysis allow us to focus on the frequency domain without being distracted by phases and low-frequency components. The spectrograms in Figure 1(b) and Figure 1(f) clearly show how the distribution of Doppler frequency changes over time, which is directly connected to changes in the movement speed. However, matching spectrograms is challenging since gesture speed changes introduce both frequency shifts and gesture



(a) Sample A (writing W, 1.5 seconds). (b) STFT result for sample A. (c) DTW over STFT for sample A. (d) DSW over STFT for sample A.

(e) Sample B (writing W, 3 seconds). (f) STFT result for sample B. (g) DTW over STFT for sample B. (h) DSW over STFT for sample B.

Figure 1. Two samples of writing W and corresponding matching results with different methods.

duration changes. For example, the slower gesture sample B in Figure 1(f) has a smaller frequency variation that lasts for a long time in the spectrogram. It is challenging to find the right scaling factor in both time and frequency domain to match the spectrogram of sample A with sample B. Figure 1(c) and 1(g) show the stretched results when we directly apply DTW on the STFT spectrogram. We observe that the DTW algorithm does not scale and match the right stages for sample A and B. Furthermore, the speed distributions in each stage of the two results are still quite different.

Neural networks, such as CNNs, can be used in classifying spectrograms with different scaling factors when trained by a large number of samples [7], [35]. However, the CNN does not consider underlying physical models of gesture spectrogram so that the training samples should exhaustively cover all speed and duration combinations of the same gesture. This incurs formidable costs in the data collection and training process.

In summary, we need to find a new nonlinear pattern matching algorithm that can compare gesture signals with different durations without the labelling information. Additionally, the algorithm needs to accommodate different speed distributions while keeping track of the movement distance.

B. Gesture speed adaptation

Before we formally define and prove the properties for gesture speed adaptation, we first use an example to show the intuition of our matching algorithm design. Our key insight is that the type of the gesture is determined by the movement trajectory of the hand rather than the movement speed. Given a certain movement stage on the trajectory, the user may move the hand at different speeds, but the different parts of the hand speed up/slow down proportionally. For example, the thumb and the index fingers move at different speeds towards each other in the *click* gesture. However, when the user clicks slowly, both fingers slow down by the same factor. Therefore, we can scale the speed distribution of a slow gesture to match with a fast gesture at the same stage. Furthermore, the stages of the gesture movement are determined by the position of hand on the trajectory. Therefore, we can track the movement stages of gestures with different speeds by cumulating the total movement distance along the trajectory. In this way, we can stretch and match the movement stages of gestures with different speeds, as shown in Figure 1(d) and Figure 1(h).

Definitions:

Let $s(t) = \sum_{p\in P} s_p(t)$ be the baseband gesture signal, where P is the set of signal propagation paths and $s_p(t)$ is the complex-valued signal along path p. We define the function $D(f,t)$, which is the square-root of the power spectral density of the gesture signal at time t:

$$D(f,t) = \left| \int_{t}^{t+\Delta t} s(\tau) e^{-j\omega\tau} \mathrm{d}\tau \right|, \tag{1}$$

where Δt is the length for a time frame. We define the trajectory that the hand moves through within Δt as a micro-unit. Consider two gesture signal samples $s_A(t)$ and $s_B(t)$ of the same gesture type. We are looking for a mapping function $\delta: \mathbb{R}\times\mathbb{R}_0^+ \to \mathbb{R}\times\mathbb{R}_0^+$ that maps the time and frequency of the gesture samples, so that $map\left(D_A(f,t),\delta\right)/D_B(f,t) = \alpha(t)$, where $\alpha(t)$ is a constant factor that only depends on time.

Assumptions:

- We use the generalized coordinates $\boldsymbol{r}$ to denote the location of different parts of the hand, *e.g.*, $\boldsymbol{r}$ is the coordinates of each finger and the palm concatenated into a single vector. Thus, different parts of the hand may go along different trajectories in the same gesture. In this section, we first consider the ideal case where the trajectories of the same gesture are exactly same so that the trace of each part is fixed. This assumption is relaxed in Section IV.
- We assume that Δt is small so that the hand moves for a short distance during one micro-unit. We further assumes that the signal amplitude and the movement speed is not changed for one micro-unit. In practice, we set Δt to 40 milliseconds so that the hand only moves for less than 4 cm in one micro-unit. So, this assumption is reasonable for real-world applications. All of the gesture signals are treated as continuous signals during the following problem description and proofs.

Speed Adaptation Properties:

Property 1. *For two signal samples $s_A(t)$ and $s_B(t)$ of the same gesture type, consider the single micro-unit from*

location $\boldsymbol{r_s}$ to $\boldsymbol{r_t}$, where the movement period for sample A is $[t_A, t_A + T_A]$ and for sample B is $[t_B, t_B + T_B]$. Without loss of generality, we assume $T_A \leq T_B$.

Then, the two spectrograms $D_A(f,t)$ and $D_B(f,t)$ satisfy the following relationship:

$$D_A(f, t_A) = \alpha D_B(\alpha f, t_B), \tag{2}$$

where α ($\alpha = \frac{T_A}{T_B} \leq 1$) is called scaling ratio of the current speed distribution.

Proof. We first consider a single part h of the hand, *e.g.*, the index finger, that is moved by a small distance of d_h from location $\boldsymbol{r_s}$ to $\boldsymbol{r_t}$. By our assumption that the movement speed is constant for the short time duration, we have the movement speed of h, $v_{h,A} = d_h/T_A$, in sample A and $v_{h,B} = d_h/T_B$ in sample B. Therefore, we have $v_{h,A} = v_{h,B}/\alpha$, where $\alpha = T_A/T_B$.

In the gesture signal, suppose there is a path p corresponding to the reflection from part h of the hand[1]. Using the signal models in [3], we have:

$$s_p(t) = A_p e^{-j(\omega v_h t/c + \theta_p)}, \tag{3}$$

where ω is the carrier frequency of the passband signal. As sample A and sample B start from the same location, we can use the same A_p and θ_p in Eq. (3).

Now consider the Fourier transform of gesture signal in sample A, $S_{p,A}(f,t) = \mathcal{F}\{s_{p,A}(t)\}$:

$$\begin{aligned} S_{p,A}(f, t_A) &= \mathcal{F}\left\{A_p e^{-j(\omega v_{h,A}(t_1 - t_A)/c + \theta_p)}\right\} \\ &= \mathcal{F}\left\{A_p e^{-j(\omega v_{h,B}((t_2 - t_B)/\alpha)/c + \theta_p)}\right\} \\ &= \alpha S_{p,B}(\alpha f, t_B), \end{aligned} \tag{4}$$

where $t_1 \in [t_A, t_A + T_A]$ and $t_2 \in [t_B, t_B + T_B]$. In the last step, we use the time scaling property of the Fourier transform, $\mathcal{F}\{x(kt)\} = \frac{1}{|k|}X(f/k)$, where the capitalized function $X(f)$ is the Fourier transforms of the non-capitalized function $x(t)$.

As Eq. (4) holds for all different parts of the hand, we can use the linearity of continuous-time Fourier transform $\mathcal{F}\{ax(t) + by(t)\} = aX(f,t) + bY(f,t)$, to get:

$$\begin{aligned} D_A(f, t_A) &= \left|\mathcal{F}\left\{\sum_{p \in P} s_{p,A}(t_A)\right\}\right| = \left|\sum_{p \in P} S_{p,A}(f, t_A)\right| \\ &= \left|\sum_{p \in P} \alpha S_{p,B}(\alpha f, t_B)\right| = \alpha D_B(\alpha f, t_B) \end{aligned} \tag{5}$$

□

Property 2. *Consider that two gesture signal samples A and B of the same gesture type. We can divide the entire trajectory into micro-units $u_1, u_2, \cdots$. Assume that the time that sample A passes through these micro-units in turn is $t_{A1}, t_{A2}, \cdots$ while the time for sample B is $t_{B1}, t_{B2}, \cdots$. There exists a mapping function $\delta : \mathbb{R} \times \mathbb{R}_0^+ \to \mathbb{R} \times \mathbb{R}_0^+$ that $map(D_A(f, t_A), \delta) = D_A(f/\alpha(t_A), t_A)$, where $\alpha(t)$ is the scaling factor function. Then the mapping function satisfies the following relationship:*

$\forall i \in N^*$, if $t_{Ai-1} \leq t \leq t_{Ai}$, $\exists\ \alpha(t) = \frac{t_{Ai} - t_{Ai-1}}{t_{Bi} - t_{Bi-1}}$,

$$map(D_A(f,t), \delta) = D_A(f/\alpha(t), t) = \alpha(t) D_B(f, t^{'}), \tag{6}$$

where $t^{'}$ is in $(t_{Bi-1}, t_{Bi}]$[2].

Proof. Consider the ith micro-unit u_i of the trajectory. According to property 1, there exists $\alpha_0 = (t_{Ai} - t_{Ai-1})/(t_{Bi} - t_{Bi-1})$ and we can scale the distribution of sample A at α_0 to get the same distribution as sample B. So,

$$D_A(f/\alpha_0, t) = \alpha_0 D_B(f, t^{'}) \tag{7}$$

Note that every micro-unit is small enough that the distribution in frequency domain is unchanged during this interval. From the above, we can get the complete expression of the scaling ratio, which is

$$\alpha(t) = \alpha_0 = \frac{t_{Ai} - t_{Ai-1}}{t_{Bi} - t_{Bi-1}},\ t \in (t_{Ai-1}, t_{Ai}],\ i \in N^*. \tag{8}$$

□

Property 1 and 2 show that we can achieve speed alignment between different samples by dynamically scaling the frequencies while ensuring that the aligned segments represent the same trajectory. Note that we can find the exact match between frequency distributions when the hand precisely follows the trajectory $\boldsymbol{r}(t)$. In reality, the hand may deviate from the trajectory $\boldsymbol{r}(t)$ so that our goal is to minimize the difference between the matched frequency distributions.

IV. Dynamic Speed Warping

In this section, we design an optimization algorithm for measuring similarity of gesture signals based on speed adaptation properties derived in Section III. Our optimization problem considers small variations in gesture trajectories so that our objective is to minimize the difference between gesture spectrograms instead of finding the exact mapping functions as in Section III. We then present a dynamic programming algorithm, which is similar to the DTW algorithm, to find the optimal solution that satisfies both the *speed* and the *cumulative movement distance* constraints. Finally, we show the micro-benchmark of this similarity measure on gesture signals and discuss the impact of parameters for the algorithm.

A. Problem Formulation

In reality, the gesture spectrogram is discretized in both the time and the frequency domain. Given two spectrograms $\boldsymbol{X}[n], n = 1 \ldots N$ and $\boldsymbol{Y}[m], m = 1 \ldots M$, where $\boldsymbol{X}[n]$ is the spectral of time-frame n. Consider a discrete warping function $F[k] = c(k), k = 1 \ldots K$, where $c(k) = (i(k), j(k))$, such that maps the spectral of $\boldsymbol{X}[i(k)]$ onto that of $\boldsymbol{Y}[j(k)]$ at the k^{th} warping. Here, $i(k)$ and $j(k)$ represents the frame index of X and Y, respectively. The warping function should be:

[1]If there are more than one path corresponds to a single hand component, we can treat each path as a separate component with a different path speed and the above result still holds.

[2]Note that $\alpha(t)$ can be greater than 1 here, which means that in the current micro-unit, the sample B is mapped to the sample A at the scaling ratio of $1/\alpha(t)$.

monotonic, *i.e.*, $i(k-1) \leq i(k)$, $j(k-1) \leq j(k)$, *continuous*, *i.e.*, $i(k) - i(k-1) \leq 1$, $j(k) - j(k-1) \leq 1$, and meeting the *boundary conditions*, *i.e.*, $i(1) = 1, j(1) = 1$, $i(K) = N, j(K) = M$ as in traditional DTW [20].

We define the scaling operation $\alpha(k)$ based on Property 1 to perform speed adaption. For example, performing an $\alpha(k)$-times scaling on $\boldsymbol{X}[i(k)]$ means linearly stretching its spectral in the frequency domain by $1/\alpha(k)$ times while shortening duration of this frame to $\alpha(k)$ in the time domain. In the following discussion, we always have $\alpha(k) < 1$ and only scale one of the $\boldsymbol{X}$ or $\boldsymbol{Y}$. We define the indicator function $I(k)$ to represent the object of the scaling operation:

$$I(k) = \begin{cases} 0 & \text{Scale } \boldsymbol{X}[i(k)], \\ 1 & \text{Scale } \boldsymbol{Y}[j(k)]. \end{cases} \tag{9}$$

Then, we define the distance between two spectral vectors $\boldsymbol{X}[i(k)]$ and $\boldsymbol{Y}[j(k)]$ after scaling as:

$$\begin{aligned} d(I, c, \alpha, k) = I(k) \left\{1 - \cos\left\langle \boldsymbol{X}[i(k)], \boldsymbol{Y}_{\alpha(k)}[j(k)] \right\rangle\right\} \\ + (1 - I(k)) \left\{1 - \cos\left\langle \boldsymbol{X}_{\alpha(k)}[i(k)], \boldsymbol{Y}[j(k)] \right\rangle\right\} \end{aligned} \tag{10}$$

where $cos\left\langle \boldsymbol{X}, \boldsymbol{Y} \right\rangle = (\boldsymbol{X} \cdot \boldsymbol{Y}) / (\|\boldsymbol{X}\| \, \|\boldsymbol{Y}\|)$ represents the cosine distance between vector $\boldsymbol{X}$ and $\boldsymbol{Y}$. The objective of DSW is to find the optimal F, α and I that minimize the average distance after the warping operation:

$$DSW(\boldsymbol{X}, \boldsymbol{Y}) = \min_{F, \alpha, I} \left[\frac{\sum_{k=1}^{K} d(I, c, \alpha, k) w(k)}{\sum_{k=1}^{K} w(k)} \right]. \tag{$*$}$$

The weight $w(k)$ is a normalizer for different time durations, where we use the symmetric form $w(k) = (i(k) - i(k-1)) + (j(k) - j(k-1))$ so that $\sum_{k=1}^{K} w(k) = N + M$.

Our optimization needs to satisfy the timing constraint so that gesture stages can be aligned. Since the scaling operation not only changes frequency distributions but also changes the time of frames, we define the frame time of $\boldsymbol{X}[i(k)]$ and $\boldsymbol{Y}[j[k]]$ after the k^{th} warping:

$$T_{i(k)} = \begin{cases} (1 - I(k)) \cdot \alpha(k) & I(k) = 0, w(k) = 2 \\ 1 & I(k) = 1, w(k) = 2 \\ (1 - I(k)) \cdot \alpha(k) & I(k) = 0, w(k) = 1 \\ 0 & I(k) = 1, w(k) = 1 \end{cases} \tag{11}$$

When we choose to scale $\boldsymbol{X}[i(k)]$ by $\alpha(k)$, whatever the $c(k-1)$ is, the time duration of $\boldsymbol{X}[i(k)]$ is scaled to $\alpha(k)$. However, if we don't scale $\boldsymbol{X}[i(k)]$, there are two cases. If $\boldsymbol{X}[i(k)]$ has been matched by $\boldsymbol{Y}[j(k-1)]$, which is equivalent to $i(k) = i(k-1)$, then we set $T_{i(k)} = 0$. Otherwise, we set $T_{i(k)} = 1$. The above piecewise function of $T_{i(k)}$ can be rewritten as:

$$T_{i(k)} = I(k) \cdot (w(k) - 1) + (1 - I(k)) \cdot \alpha(k) \tag{12}$$

By symmetry, we have $T_{j(k)}$ as shown in Eq. (13).

$$T_{j(k)} = (1 - I(k)) \cdot (w(k) - 1) + I(k) \cdot \alpha(k) \tag{13}$$

Therefore, the timing constraint can be expressed as:

$$\left| \sum_{k=1}^{K} T_{i(k)} - \sum_{k=1}^{K} T_{j(k)} \right| \leq Q, \tag{$**$}$$

Algorithm 1: Basic Dynamic Speed Warping

Input: Two spectrograms $\boldsymbol{X}[n], n = 1, \cdots, N$ and $\boldsymbol{Y}[m], m = 1, \cdots, M$, scaling ratio list $\boldsymbol{\alpha}[v], v = 1, \cdots, V$.
Output: $\min(d_{DSW}[N, M, v, \mu]), v = 1, \cdots, V, \mu = 0, 1$.
// $d_{DSW}[n, m, v, \mu]$: 4d array recording distance between two spectrograms at each middle state.
// $l_s[n, m, v, \mu]$: 4d array recording differences of duration for scaled $\boldsymbol{X}$ and $\boldsymbol{Y}$ from initial to current state.
/* Initialization. */
for $n = 1, \ldots, N$, $m = 1, \ldots, M$, $v = 1, \ldots, V$ **do**
  $d_{DSW}[n, m, v, \mu] \leftarrow \infty$
  $d_{DSW}[0, 0, v, \mu] \leftarrow 0$
  $l_s[n, m, v, \mu] \leftarrow 0$
end
for $n = 1, \ldots, N$, $m = 1, \ldots, M$, $v = 1, \ldots, V$ **do**
  /* Scale $\boldsymbol{X}[n]$ */
  if $\mu = 0$ **then**
    // q_1, q_2: candidate index sets
    // d_1, d_2: distance of alternative paths.
    $dist = 1 - cos\langle \boldsymbol{X}_{\boldsymbol{\alpha}[v]}[n], \boldsymbol{Y}[m] \rangle$
    /* Case 1: $(n-1, m) \rightarrow (n, m)$ */
    $q_1 \leftarrow \{(n-1, m, v, 0) \mid |l_s[n-1, m, v, 0]| \leq Q\}$;
    $d_1 \leftarrow min(\{d_{DSW}[n-1, m, v, 0] \mid (n-1, m, v, 0) \in q_1\})$;
    /* Case 2: $(n-1, m-1) \rightarrow (n, m)$ */
    $q_2 \leftarrow \{(n-1, m-1, v, \mu^{'}) \mid |l_s[n-1, m-1, v, \mu^{'}]| \leq Q\}$;
    $d_2 \leftarrow min(\{d_{DSW}[n-1, m-1, v, \mu^{'}] \mid (n-1, m-1, v, \mu^{'}) \in q_2\})$;
    $d_{DSW}[n, m, v, 0] \leftarrow min(d_1 + dist,\ d_2 + 2\,dist)$
    Update l_s according to equation 16.
  end
  /* Scale $\boldsymbol{Y}[m]$ */
  else
    $d_{DSW}[n, m, v, 1]$ can be calculated in a similar way as $d_{DSW}[n, m, v, 0]$.
  end
end
for $v = 1, \ldots, V$ *and* $\mu = 0, 1$ **do**
  $d_{DSW}[N, M, v, \mu] \leftarrow d_{DSW}[N, M, v, \mu]/(N + M)$
end
return $\min(d_{DSW}[N, M, v, \mu])$, $v = 1, \cdots, V$, $\mu = 0, 1$.

Here $\sum_{k=1}^{K} T_{i(k)}$ represents sum of the *scaled* duration from $\boldsymbol{X}[i(1)]$ to $\boldsymbol{X}[i(k)]$ and Q is the threshold of duration differences that decides suitable candidate warping functions.

B. Basic Dynamic Speed Warping

In this section we first present a dynamical programming algorithm to solve the above optimization, which serves as a basic version of our final solution. We use a four-dimensional array d_{DSW} to maintain the smallest dissimilarity between the matching part of $\boldsymbol{X}$ and $\boldsymbol{Y}$ at each intermediate state. In the array $d_{DSW}[n, m, v, \mu]$, the first and the second index, n and m, are the current time-frame index for $\boldsymbol{X}$ and $\boldsymbol{Y}$. The third index v indicates the scaling factor for the matching and the fourth index $\mu = 0$, 1 is the indication function for whether scaling $\boldsymbol{X}$ or $\boldsymbol{Y}$. Thus, $d_{DSW}[n, m, v, 0]$ is the shortest distance between the matched part of two spectrograms when $\boldsymbol{X}[n]$ is scaled by $\boldsymbol{\alpha}[v]$ times and then matched with $\boldsymbol{Y}[m]$. We use a searchable scaling ratio array $\alpha[v], v = 1, \cdots, V$ for convenience. We also maintain another four-dimensional array l_s, which tracks the difference of duration for the matched part of $\boldsymbol{X}$ and $\boldsymbol{Y}$ to meet the constraint in Eq. (**).

The basic DSW algorithm is shown in Algorithm 1. Suppose the k^{th} warping of warping function F is $c(k) = (n, m)$. The monotonicity and continuity of F determine that the previous match $c(k-1)$ can only come from the following three cases: $(n-1, m)$, $(n, m-1)$ or $(n-1, m-1)$. After considering the scaling operation, the transfer process from $c(k-1)$ to $c(k)$ is related to $\boldsymbol{\alpha}(k)$, too. Take scaling $\boldsymbol{X}[n]$ as an example, which is equivalent to $\mu = 0$. There are only two cases for $c(k-1)$:

$$c(k-1) = \begin{cases} (n-1, m) & \mu^{'} = 0 \\ (n-1, m-1) & \mu^{'} = 0 \ or \ 1. \end{cases} \tag{14}$$

where $\mu^{'}$ indicates whether to scale $\boldsymbol{X}$ or $\boldsymbol{Y}$ at the $(k-1)^{th}$ warping. Note that $c(k-1)$ cannot be $(n, m-1)$, because each frame can be scaled by at most once during the whole warping progress. If we scale $\boldsymbol{X}[n]$ at the $(k-1)^{th}$ warping, then $\boldsymbol{X}[n]$ is matched with both $\boldsymbol{Y}[m-1]$ and $\boldsymbol{Y}[m]$. It is impossible to match a shortened $\boldsymbol{X}[n]$ to two frames. If we choose to scale $\boldsymbol{Y}[m-1]$, it's equivalent to scale $\boldsymbol{X}[n]$ at two different ratios $1/\boldsymbol{\alpha}[k-1]$ and $\boldsymbol{\alpha}[k]$, which is also impossible.

During the warping process, we need to track whether the two cumulative duration before the k^{th} warping is close enough after a series of scaling operations. So, we use q_1 and q_2 to limit the difference between two cumulative duration, and then select the minimum distance d_1 and d_2 among the set of indexes that satisfy the timing constraint. We can update $d_{DSW}\ [n, m, v, 0]$ as:

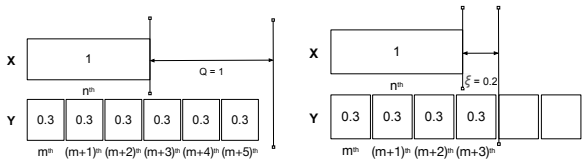
$$d_{DSW}[n, m, v, 0] = \min \begin{cases} \min\{d_{DSW}[n-1, m, v^{'}, 0] + dist\} \\ \quad (n-1, m, v^{'}, 0) \in q_1 \\ \min\{d_{DSW}[n-1, m-1, v^{'}, \mu^{'}] + 2\,dist\} \\ \quad (n-1, m, v^{'}, \mu^{'}) \in q_2 \end{cases} \tag{15}$$

where we use the symmetric form of weight $w(k)$, *i.e.*, $w(k)$ is 1 for case 1 and 2 for case 2. Then, l_s is updated as:

$$l_s[n, m, v, 0] = \begin{cases} l_s[n-1, m, v^{'}, 0] + \alpha[v^{'}] \\ \quad \text{updated with } (n-1, m, v^{'}, 0); \\ l_s[n-1, m-1, v^{'}, \mu] + \alpha[v^{'}] - 1 \\ \quad \text{updated with } (n-1, m-1, v^{'}, \mu). \end{cases} \tag{16}$$

Symmetrically, assume that $\boldsymbol{Y}[j]$ is scaled by $\alpha[k]$ times. We use a similar method to divide the transfer process into two cases to calculate $d_{DSW}[n, m, v, 1]$. Finally, the distance between the two spectrograms is determined by the minimum of all possible final states.

The DSW algorithm has the optimal substructure and meets the requirement of the dynamical programming algorithm. Given an optimal warping function F, which uniquely determines a transfer route of the four-dimensional array d_{DSW} from $s = (0, 0, 0, 0)$ to $t = (N, M, v_t, \mu_t)$. Let $u = (n, m, v, \mu)$ be an intermediate state of F, the transfer route p from s to u should also be the shortest. This can be proved by contradiction. Suppose there is a new transfer route p_1 such that $d_{DSW \rightsquigarrow_{p_1}}[n, m, v, \mu] \leq d_{DSW \rightsquigarrow_p}[n, m, v, \mu]$, and let p_2 be the transfer route from u to t. If the transfer route $s \rightsquigarrow_{p_1} u \rightsquigarrow_{p_2} t$ can meet the timing constraint during the whole progress, then $d_{DSW \rightsquigarrow_{p_1} \rightsquigarrow_{p_2}}(N, M, v_t, \mu_t) \leq d_{DSW \rightsquigarrow_p \rightsquigarrow_{p_2}}(N, M, v_t, \mu_t)$, which is in contradiction with $s \rightsquigarrow_p u \rightsquigarrow_{p_2} t$ being the optimal transfer route. Otherwise, u cannot be in the intermediate state of F, which is also contrary to the assumptions.



(a) Matching of basic DSW. (b) Matching of refined DSW.

Figure 2. Problem of basic DSW algorithm and our improvement.

In this way, the DSW algorithm provides a new measure of dissimilarity that satisfies the following properties:

$$0 = DSW(\boldsymbol{X}, \boldsymbol{X}) < DSW(\boldsymbol{X}, \boldsymbol{Y}) = DSW(\boldsymbol{Y}, \boldsymbol{X}). \tag{17}$$

It is easy to verify that the DSW distance is non-negative and $DSW(\boldsymbol{X}, \boldsymbol{X}) = 0$. In addition, the symmetry of DSW can be proved by induction, which we will not elaborate here.

The time complexity of the basic DSW algorithm is $O(VNMN_f)$, where the numbers of frames in the spectrograms are N and M, the number of samples in the frequency domain is N_f, and the size of the scaling ratio list is V. As the number of candidate scaling ratio V is a constant, the time complexity of DSW is a constant factor to the complexity of DTW algorithms over spectrograms, which is $O(NMN_f)$. Note that the interpolation operation for scaling the frequency distribution has been completed in the data preprocessing stage, so the time cost of interpolation is not considered here.

C. Refined Dynamic Speed Warping

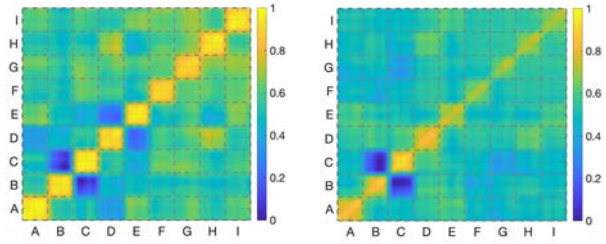
The basic DSW algorithm may match an excessive amount of frames to a scaled frame. Consider the situation shown in Figure 2. Two different spectrograms $\boldsymbol{X}$ and $\boldsymbol{Y}$ of the same gesture are matched using the basic DSW algorithm, while the timing constraint Q is set to the duration of one frame. With the basic DSW, a single frame of the slower gesture $\boldsymbol{X}[n]$ can be matched to five scaled frames of $\boldsymbol{Y}[n]$, given that $|\sum_{j=m}^{m+5} T_j - T_n|$ does not exceed the threshold Q. However, this leads to distortions in the warping function, as the movement distance along the trajectory of the frame $\boldsymbol{X}[n]$ is closer to frames $\boldsymbol{Y}[m]$ to $\boldsymbol{Y}[m+3]$ and the rest frames should be matched the next frame $\boldsymbol{X}[n+1]$. To eliminate the distortion caused by such repeated matching, we impose a new constraint on the timing of frames.

We define the matching set S for each $\boldsymbol{X}[i]$ for a specific warping function F as:

$$\begin{aligned} S_{\boldsymbol{X}[i]} = \{\boldsymbol{Y}[j] \mid \exists\ k \leq K \text{ and } k \in N^+, \\ c(k) = (\boldsymbol{X}[i], \boldsymbol{Y}[j]) \text{ and } I(k) = 1\} \end{aligned} \tag{18}$$

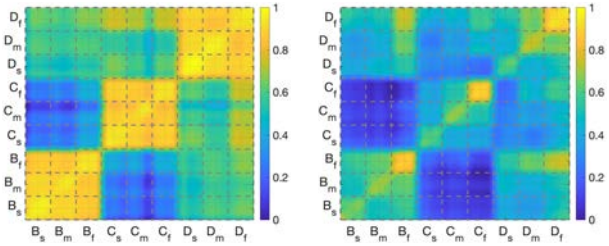
This set represents all frames $\boldsymbol{Y}[j]$ in warping function F that matched $\boldsymbol{X}[i]$. Symmetrically, $S_{\boldsymbol{Y}[j]}$ can be defined as in Eq. (18). The tighter movement distance constraint is formalized as:

$$\forall \boldsymbol{z} \in \boldsymbol{X} \bigcup \boldsymbol{Y} \text{ and } |S_{\boldsymbol{z}}| \geq 0, \ \sum_{\boldsymbol{z}^{'} \in S_{\boldsymbol{z}}} T_{\boldsymbol{z}^{'}} - 1 \leq \xi. \quad (***)$$



(a) Confusion matrix baseline of DSW (normalized). (b) Confusion matrix baseline of DTW (normalized).

Figure 3. Gesture recognition capability.



(a) Confusion matrix of DSW on different speeds (normalized). (b) Confusion matrix of DTW on different speeds (normalized).

Figure 4. Capability of speed adaptation.

The above equation shows that the total movement distance on frames $\boldsymbol{z}^{'}$ in $S_{\boldsymbol{z}}$ should be close to that of the matched $\boldsymbol{z}$, which we consider as a micro-unit in Property 1.

With the tighter movement distance constraint, the original time constraint in Eq. (**) can also be redefined. Define an ordered set $U = \{(\boldsymbol{x}, S_{\boldsymbol{x}})|\forall \boldsymbol{x} \in \boldsymbol{X}, |S_{\boldsymbol{x}}| \geq 1\} \bigcup \{(\boldsymbol{y}, Q_{\boldsymbol{y}})|\forall \boldsymbol{y} \in \boldsymbol{Y}, |Q_{\boldsymbol{y}}| \geq 1\}$ that represents the frames matched on the same micro-unit. Then, the time difference of the same mirco-unit $U[l]$ is

$$\xi[l] = \begin{cases} \sum\limits_{\boldsymbol{y} \in S_{\boldsymbol{x}}} T_{\boldsymbol{y}} - 1 & U[l] = (\boldsymbol{x}, S_{\boldsymbol{x}}) \\ 1 - \sum\limits_{\boldsymbol{x} \in S_{\boldsymbol{y}}} T_{\boldsymbol{x}} & U[l] = (\boldsymbol{y}, S_{\boldsymbol{y}}) \end{cases} \tag{19}$$

Based on the above definition, the timing constraint Eq. (**) can be re-written as follows:

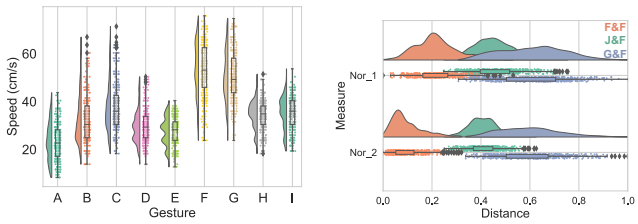
$$\forall l \in N^* \text{ and } l \in [1, L], \left| \sum_{v=1}^{l} \xi[v] \right| \leq Q, \qquad (****)$$

where Q represents the max cumulative timing error of the first l micro-unit after scaling.

In the refined DSW algorithm, we use a new four-dimensional array l_c to record the total duration of elements in a matching set. Consider the k^{th} warping state where $\boldsymbol{X}_{\alpha[v]}[n]$ and $\boldsymbol{Y}[m]$ are matched. If $c(k-1) = (n-1, m)$, both $\boldsymbol{X}[n-1]$ ad $\boldsymbol{X}[n]$ are in the matching set of $\boldsymbol{Y}[m]$. In this case, we need to modify q_1 as $\{(n-1, m, v^{'}, 0) \mid l_c[n-1, m, v^{'}, 0] \leq 1\}$. If $c(k-1) = (n-1, m-1)$, the matching starts with a new micro-unit since $c(k) = (m, n)$, so no change is required. The refined algorithm avoids the distortion in the warping and eliminates redundant calculations for the timing constraint. In the following discussions, the DSW algorithm refers to the refined DSW algorithm when not specifically mentioned.

D. Discussion and Parameter Selection

We first use a small set of ultrasound gesture signals collected by smartphones to demonstrate the performance of



(a) Speed distribution of gestures. (b) Different normalization methods.

Figure 5. Selecting hyperparameters and normalization measurements.

DSW. The data set is 225 gesture samples of nine gestures performed by a single user at a smooth speed. The distances results of DSW are shown in Figure 3(a). It can be seen that as a similarity measure, the DSW algorithm not only accurately recognizes similar gestures but also well distinguishes different types of gestures. In contrast, DTW-based similarity measure cannot guarantee the synchronization of cumulative movement distances. Therefore, the similarity of complex gestures, such as F∼I, could be significantly underestimated by DTW without tracking the movement progress, as shown in Figure 3(b).

To further understand how DSW adapts to different speeds, we use a special gesture set that has three gesture types (Pushing near, Pushing away, and Circling clockwise) with 25 gesture samples performed at three different speeds (fast, medium, and slow). The distance results of DSW and DTW are shown in Figure 4. We observe from Figure 4(a) that DSW accurately classifies the gestures regardless of their difference in speeds. However, DTW does not handle the change of speeds very well and only gestures with similar speeds can be recognized.

To fully explore the capability of DSW, we need to carefully design the hyperparameters of the DSW algorithm, including the minimum speed scaling ratio α_{min}, the length of scaling ratio list V and the time constraint threshold Q. The minimum speed scaling ratio is selected based on how fast/slow that users will perform the gesture. We first collect gesture samples of ten volunteers, who repeat these gestures at the speed that they feel comfortable. We then determine the peak speed of each gesture sample and derive the speed distribution of different gesture types, as in Figure 5(a). We observe that the highest gesture speed can reach $75\,cm/s$, and the speed resolution of gestures is $6\,cm/s$. Thus, in order to satisfy all possible speed adaptation requirements, we set $\alpha_{min} = 0.1$ so that a suitable warping function can be found for two samples with a maximum speed difference of ten times.

Our experimental results show that different lengths of the scaling ratio list V and the different thresholds of the time constraint Q have little effect on the distance calculation. However, the complexity of DSW is significantly reduced with a shorter list of $\boldsymbol{\alpha}$. At the same time, a tighter threshold can effectively prune warping paths, which also speeds up the execution of the algorithm. In following discussions, we set $V = 6$ and $Q = 1$ so that the theoretical execution time of DSW is $2V = 12$ times of the DTW algorithm. Our experiments show DTW takes 0.075 seconds to compare two gesture signals with a length of one second. After applying

Table I
LIST OF GESTURES

Gesture	Label	Gesture	Label
Click	A	Pushing Near	B
Pushing Away	C	Circling Clockwise	D
Circling anti-clockwise	E	Palm drawing N	F
Palm drawing inverted N	G	Writing W	H
Writing M	I		

a pruning operation for DSW, the average execution time for DSW is 0.55 seconds, which is 7.3 times that of DTW. Therefore, DSW can effectively compare gestures in real-time.

Another important design consideration is to choose the data-normalization measures for spectral vectors in the DSW. We compare the performance of two common normalization methods, min-max normalization (Nor_1) and scaling to unit length (Nor_2) in Figure 5(b). We observe that scaling to unit length leads to better separation between different gestures, *i.e.*, less overlapping between gestures. Therefore, we choose to use the second data-normalization method in DSW.

V. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of DSW using the gesture dataset collected through commercial mobile phones. We first quantitatively evaluate the performance of DSW for the one-shot learning scenario and compare it to traditional methods such as DTW, SVM, and CNN. We then explore the efficiency of DSW in the unsupervised learning scenario for gesture clustering tasks.

A. Dataset Description

We implement the ultrasound-based gesture sensing system on the Android platform. The gesture signals are collected using a Samsung Galaxy S7 mobile phone in a lab environment. We first emit sinusoid signals with a frequency of 18 kHz using the speaker on the smartphone. We then use the microphone on the smartphone to record the hand reflection signal with a sampling rate of 48 kHz. The recorded signal is mixed with the transmitted sinusoids in a digital down-converter. The output of the down-converter is a complex-valued baseband signal. The sampling rate of the baseband signal is decimated by 8 times from 48 kHz down to 6 kHz.

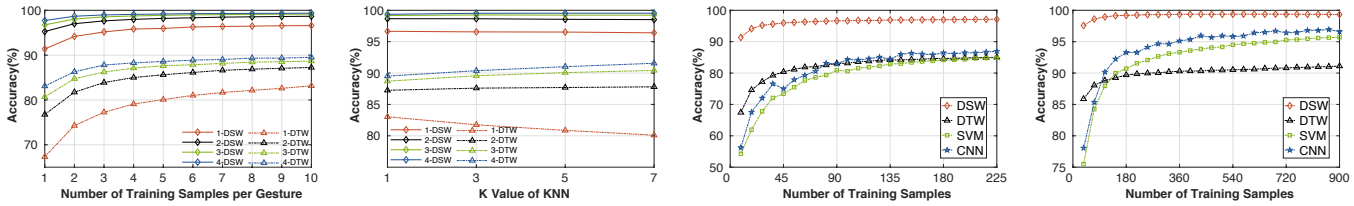
We collect 2,250 gesture samples for nine types of gestures from ten participants. The categories of gestures are shown in Table I. Each participant performs each class of the gestures 25 times at any speed they feel comfortable in the same region respective to the phone. The preprocessing for gesture samples includes the following steps. We first perform STFT on the baseband signals with a 1024-point FFT window that moves 256 points for each step. Under the above configuration, the frequency resolution of STFT is around 5.86 Hz so that it can distinguish movements with speed differences of around 6 cm/s. We then perform column-wise normalization on each STFT frame $\boldsymbol{X}$, so that the modulus of each spectral vector $|\boldsymbol{X}[n]| = 1$. It should be noted that we also save the spectral vectors that have been scaled by α times to reduce the computational cost in the later scaling steps. We keep the spectral vectors in frequency range of $[-117.19 \text{ Hz}, 117.19 \text{ Hz}]$, which is corresponding to the speed range of $[-1.2 \text{ m/s}, 1.2 \text{ m/s}]$.

B. Effect on One-shot Learning Tasks

Experimental results show that the Nearest Neighbour (1NN) with DSW algorithm achieves a recognition accuracy of 99.47% when both training set and testing set are from the same domain. We randomly select 1,125 samples as testing set from our gesture samples, ensuring that it contains different samples of the same participant. Then, we randomly select just one sample of each gesture type for each of the ten participants to serve as the training set. Thus, the training set contains 90 gesture sample. With 200 rounds of Monte Carlo cross-validation, we find that the classification error rate of DSW-1NN is always less than 0.53%. This coincides with the common understanding that DTW-like algorithms performs well even with the simple 1NN.

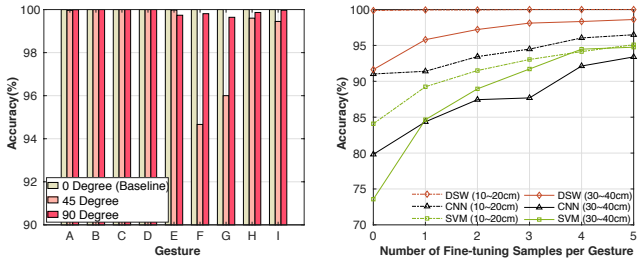
We then compare the generalization performance in a more challenging scenario where testing samples are from a different domain. In this experiment, we select gesture data from one to four users as the training set and the data from the remaining six users constitutes the testing set. We first compare the performance of two memory-based classifiers, kNN based on the DSW algorithm (DSW-kNN) and kNN based on the DTW algorithm (DTW-kNN). As shown in Figure 6(a), when the training data contains only one user, the average accuracy of the DSW-1NN on the testing set for other users increase from 91.38% to 96.58% when we use one to ten samples of each gesture type for training. At the same time, as the number of trainers increases to four, the accuracy rate raises to a maximum of 99.36%. However, with the same amount of training data, the DTW-1NN has a maximum accuracy of 89.52%. The result shows that the DSW-kNN can achieve an accuracy of no less than 97% with a minimum of one gesture sample per user when applied to different domains. Figure 6(b) shows us the accuracy of kNN based on two different distance methods as k increases. It can be found that the accuracy of DSW-kNN does not change with the value of k, which means that the distance density of the same type for DSW does not vary with the number of samples. However, the accuracy of DTW-kNN increases with the increase of k value, indicating that the similar samples obtained by DTW are sparse when the size of the dataset is small.

We further compare the DSW-1NN with the feature-based classifier, such as CNNs and Support Vector Machines (SVM) in Figure 6(c) and Figure 6(d). Here, we choose the classic LeNet structure [36] for CNN and linear kernel for SVM. We observe that CNN and SVM rely on a large amount of training data and the diversity of participants. When the training set is small, they may learn too many irrelevant features and overfit the dataset. Figure 6(c) shows that with only one user's training data, the DSW-1NN has a strong generalization capability that can achieve an accuracy of 97.19%, while the other three methods have an accuracy of no more than 86%. In addition, in the case that the training set contains four participants, DSW-1NN only needs 36 template samples to achieve 97.58% generalization accuracy, which is higher than the CNN model based on 900 samples. When using 900 samples, the maximum



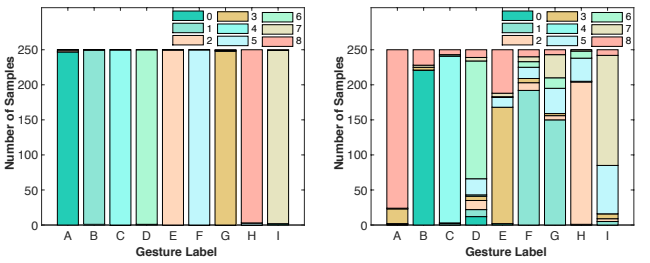
(a) DSW-1NN vs DTW-1NN. (b) DSW-kNN vs DTW-kNN. (c) Methods on one trainer's data. (d) Methods on four trainers' data.

Figure 6. Generalization capabilities of different supervised learning methods.



(a) Different angles. (b) Different distances.

Figure 7. Robustness and fine-tuning of DSW-1NN.



(a) DSW-based AP clustering. (b) DTW-based AP clustering.

Figure 8. Clustering results based on DSW or DTW.

Table II
VALIDITY INDEX OF CLUSTERING

Index	DSW	DTW
JC	0.9772	0.4380
FMI	0.9884	0.6094
RI	0.9974	0.9110

generalization accuracy of DSW-1NN, DTW-1NN, CNN and SVM is 99.33%, 91.11%, 96.61% and 95.70%, respectively.

DSW-1NN is also robust to gestures at different angles and different distances. To evaluate the influence of different angles on the model, we request a volunteer to perform gestures at three angles (0, 45, and 90 degree with respect to the center of the phone) while maintaining an equal distance to the phone. We use the 0-degree gesture samples as the training set, which has five samples for each gesture type. Then, we evaluate the performance on test datasets at three different angles, each contains 225 gesture samples. After 50 rounds of Monte Carlo cross-validation, we find that the accuracy of DSW-1NN at different angles is 100%, 98.85%, and 99.89%, respectively. We also request the volunteer to perform gestures at three different distances (10∼20 cm, 20∼30 cm and 30∼40 cm) from the mobile phone. We use the 45 samples in the 20∼30 cm region as the source domain and DSW-1NN has an accuracy of 99.86% and 91.64% on the testing sets of 10∼20 cm and 30∼40 cm. This means that the DSW-1NN algorithm is robust to distance changes within 30 cm. When the distance changes by more than 30 cm, the accuracy of the DSW-1NN algorithm is reduced to 91% due to the signal attenuation. However, we can fine-tune the model by adding a small number of samples in the target domain without retraining. As shown in Figure 7(b), after adding five target-domain samples for each gesture type, the accuracy of the 30∼40 cm region is increased to 98.61%. In the same situation, CNN and SVM can only reach 79.8% and 73.57% before fine-tuning, and the model parameters need to be retrained for the target domain.

C. Evaluation for Clustering Tasks

Clustering is another application scenario for similarity measure like DSW and DTW. The clustering algorithm based on the geometric relationship to find the cluster center is not applicable for these two distances, as they do not satisfy the triangle inequality. Thus, we use the Affinity Propagation (AP) clustering algorithm [37]. We adjust the damping coefficient and preference factor to achieve aggregation of different numbers of clusters. As we have nine gesture types, we choose to converge at nine clusters. As shown in Figure 8(a), the distance defined by DSW correctly cluster the gesture of the same type into the same category. However, the DTW-based clustering does not give the correct clustering results. We further use three common external indicators to represent cluster validity, such as Jaccard Coefficient (JC), Fowles and Mallows Index (FMI) and Rand Index (RI), where a larger indicator means better clustering performance. As shown in Table II, DSW-based clustering has higher scores for all three indicators than DTW. This proves that the similarity measure defined by DSW can be used for clustering large-scale unlabeled data and can even guide the design of gestures.

VI. CONCLUSION

In this paper, we introduce DSW, a dynamical speed adaptive matching scheme for gesture signals. DSW rescales the speed distributions of gesture samples while keeping track of the movement distance at different stages. Our experimental results show that DSW provides a robust similarity measure between gesture samples and can work for both one-shot learning in supervised gesture recognition and unsupervised learning tasks. While we mainly focus on sound-based signals and STFT features, we believe that DSW can be readily applied to Wi-Fi based gesture recognition systems and other time-frequency analysis schemes, such as DWT or HHT.

VII. ACKNOWLEDGEMENT

We would like to thank our anonymous shepherd and reviewers for their valuable comments. This work is partially supported by National Natural Science Foundation of China under Numbers 61872173 , 61972192, and Collaborative Innovation Center of Novel Software Technology.

REFERENCES

[1] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proceedings of ACM MobiCom*, 2013.

[2] F. Adib and D. Katabi, "See through walls with WiFi!," in *Proceedings of ACM SIGCOMM*, pp. 75–86, 2013.

[3] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in *Proceedings of ACM MobiCom*, 2015.

[4] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics*, vol. 35, no. 4, p. 142, 2016.

[5] T. Wei and X. Zhang, "mTrack: High-precision passive tracking using millimeter wave radios," in *Proceedings of ACM MobiCom*, 2015.

[6] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," in *Proceedings of USENIX NSDI*, 2015.

[7] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "RF-based 3D skeletons," in *Proceedings of ACM SIGCOMM*, 2018.

[8] J. Wang, H. Jiang, J. Xiong, K. Jamieson, X. Chen, D. Fang, and B. Xie, "LiFS: low human-effort, device-free localization with fine-grained subcarrier information," in *Proceedings of ACM MobiCom*, 2016.

[9] Y. Ma, G. Zhou, S. Wang, H. Zhao, and W. Jung, "SignFi: Sign language recognition using WiFi," in *Proceedings of ACM UbiComp*, 2018.

[10] W. Wang, A. X. Liu, and K. Sun, "Device-free gesture tracking using acoustic signals," in *Proceedings of ACM MobiCom*, 2016.

[11] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota, "FingerIO: Using active sonar for fine-grained finger tracking," in *Proceedings of ACM CHI*, 2016.

[12] S. Yun, Y.-C. Chen, H. Zheng, L. Qiu, and W. Mao, "Strata: Fine-grained acoustic-based device-free tracking," in *Proceedings of ACM MobiSys*, pp. 15–28, ACM, 2017.

[13] K. Ling, H. Dai, Y. Liu, and A. X. Liu, "Ultragesture: Fine-grained gesture sensing and recognition," in *Proceedings of IEEE SECON*, 2018.

[14] T. Wang, D. Zhang, Y. Zheng, T. Gu, X. Zhou, and B. Dorizzi, "C-FMCW based contactless respiration detection using acoustic signal," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, p. 170, 2018.

[15] K. Sun, W. Wang, A. X. Liu, and H. Dai, "Depth aware finger tapping on virtual displays," in *Proceedings of ACM MobiSys*, pp. 283–295, ACM, 2018.

[16] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas, *et al.*, "Towards environment independent device free human activity recognition," in *Proceedings of ACM MobiCom*, pp. 289–304, 2018.

[17] H. Du, P. Li, H. Zhou, W. Gong, G. Luo, and P. Yang, "Wordrecorder: Accurate acoustic-based handwriting recognition using deep learning," in *Proceedings of IEEE INFOCOM*, pp. 1448–1456, IEEE, 2018.

[18] A. Virmani and M. Shahzad, "Position and orientation agnostic gesture recognition using WiFi," in *Proceedings of ACM MobiSys*, 2017.

[19] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-effort cross-domain gesture recognition with Wi-Fi," in *Proceedings of ACM MobiSys*, 2019.

[20] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 43–49, 1978.

[21] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: In-home device-free activity identification using fine-grained WiFi signatures," in *Proceedings of ACM MobiCom*, 2014.

[22] C. Han, K. Wu, Y. Wang, and L. M. Ni, "WiFall: Device-free fall detection by wireless networks," in *Proceedings of IEEE INFOCOM*, pp. 271–279, 2014.

[23] H. Abdelnasser, M. Youssef, and K. A. Harras, "WiGest: A ubiquitous wifi-based gesture recognition system," in *Proceedings of IEEE INFOCOM*, 2015.

[24] B. Kellogg, V. Talla, and S. Gollakota, "Bringing gesture recognition to all devices," in *Proceedings of Usenix NSDI*, 2014.

[25] W. Ruan, Q. Z. Sheng, L. Yang, T. Gu, P. Xu, and L. Shangguan, "Audiogest: enabling fine-grained hand gesture detection by decoding echo signal," in *Proceedings of ACM UbiComp*, 2016.

[26] J. Zhang, Z. Tang, M. Li, D. Fang, P. Nurmi, and Z. Wang, "CrossSense: towards cross-site and large-scale WiFi sensing," in *Proceedings of ACM MobiCom*, 2018.

[27] J. Wang, Y. Chen, L. Hu, X. Peng, and S. Y. Philip, "Stratified transfer learning for cross-domain activity recognition," in *Proceedings of IEEE PerCom*, pp. 1–10, IEEE, 2018.

[28] L. Zhang, Z. Wang, and L. Yang, "Commercial Wi-Fi based fall detection with environment influence mitigation," in *Proceedings of IEEE SECON*, 2019.

[29] K. Chen, L. Yao, D. Zhang, X. Chang, G. Long, and S. Wang, "Distributionally robust semi-supervised learning for people-centric sensing," in *Proceedings of ACM AAAI*, 2018.

[30] 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, 2015.

[31] N. Gillian, B. Knapp, and S. O'modhrain, "Recognition of multivariate temporal musical gestures using n-dimensional dynamic time warping," in *Proceedings of NIME*, 2011.

[32] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using WiFi signals," in *Proceedings of ACM MobiCom*, 2015.

[33] H. Li, W. Yang, J. Wang, Y. Xu, and L. Huang, "WiFinger: talk to your smart devices with finger-grained gesture," in *Proceedings of ACM UbiComp*, 2016.

[34] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, "We can hear you with Wi-Fi!," in *Proceedings of ACM MobiCom*, 2014.

[35] X. Xu, J. Yu, Y. Chen, Y. Zhu, L. Kong, and M. Li, "Breathlistener: Fine-grained breathing monitoring in driving environments utilizing acoustic signals," in *Proceedings of ACM MobiSys*, pp. 54–66, ACM, 2019.

[36] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[37] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *science*, vol. 315, no. 5814, pp. 972–976, 2007.