Temporal Hierarchical Dictionary Guided Decoding for Online Gesture Segmentation and Recognition

Haoyu Chen, *Student Member, IEEE,* Xin Liu, *Member, IEEE,* Jingang Shi, *Member, IEEE,* and Guoying Zhao*, *Senior Member, IEEE*

***Abstract*—Online segmentation and recognition of skeleton-based gestures are challenging. Compared with offline cases, the inference of online settings can only rely on the current few frames and always completes before whole temporal movements are performed. However, incompletely performed gestures are ambiguous and their early recognition is easy to fall into local optimum. In this work, we address the problem with a temporal hierarchical dictionary to guide the hidden Markov model (HMM) decoding procedure. The intuition is that, gestures are ambiguous with high uncertainty at early performing phases, and only become discriminate after certain phases. This uncertainty naturally can be measured by entropy. Thus, we propose a measurement called "relative entropy map" (REM) to encode this temporal context to guide HMM decoding. Furthermore, we introduce a progressive learning strategy with which neural networks could learn a robust recognition of HMM states in an iterative manner. The performance of our method is intensively evaluated on three challenging databases and achieves state-of-the-art results. Our method shows the abilities of both extracting the discriminate connotations and reducing large redundancy in the HMM transition process. It is verified that our framework can achieve online recognition of continuous gesture streams even when they are halfway performed.**

***Index Terms*—Temporal context, hidden Markov model, hierarchical structure, deep neural network, relative entropy, skeleton-based recognition.**

I. Introduction

HUMAN gestures are ubiquitous in the visual cognition, pervading body language in all ages, cultures and tightly integrated with verbal communication [1]. As an alternative source to conventional RGB videos [2], the 3D skeletal joint coordinates obtained from e.g., Kinects, contain compact 3D positions of the major human body joints, which are robust to variations of viewpoints [3]. Thus, skeleton-based action and gesture recognition have been widely studied in recent years. Meanwhile, compared to the offline setting, online gesture segmentation and recognition can meet the low-latency requirement and has more potential in applications spanning sign-language recognition [4], virtual manipulation to daily assistance [5]. However, data-driven methods for the online skeleton-based dynamic gesture recognition are still facing several open challenges in real-world applications.

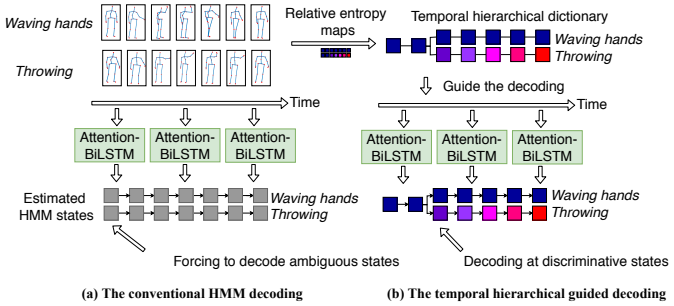

Fig. 1. Utilizing HMM for continuous gesture stream modeling. (a) Existing methods [6], [7] utilize an ordinary HMM model with all the states being candidates for transition. However, forcing to decode ambiguous HMM states with insufficient information is easy to fall into local optimum. (b) The proposed HMM transition decoding is guided by a temporal hierarchical dictionary which makes the HMM transition more discriminating and efficiently narrows down the search range during the decoding.

The first challenge of the online setting is that, the complete global information is often unavailable. It is because that the online setting requires the recognition to complete fast even before the whole gesture sequences are seen. Recently, temporal dynamic deep models [8] [9] [10] [11] [12] like recurrent neural networks (RNN) show the capability to model the temporal dependency for gesture recognition. However, their superb performances much rely on the inference on the complete gesture sequences [13]. Among those methods, the segmentation of gestures from skeleton joint sequences is often ignored under the assumption that pre-segmented sequences are available [14]. However, in the tasks of online gesture segmentation and recognition, the inference is limited within the local temporal information from the current few frames. The hidden Markov model (HMM), with its sequential temporal state modeling capability, can naturally process temporal semantic connotations in an online manner. Thus, we merge the Deep Neural Network (DNN) into an HMM prototype to propose a hybrid DNN-HMM framework for the online setting.

Secondly, human gestures are more symbolic and semantic when compared to human actions and activities [5]. At its root, the motion of a gesture is a set of sequentially distinct phases organized in a global temporal order [15]. Different gestures may have some phases highly similar. As shown in

*Corresponding author

This work is supported by the Academy of Finland for project MiGA (grant 316765), ICT 2023 project (grant 328115), and Infotech Oulu and in part by the Chinese Scholarship Council. As well, the authors wish to acknowledge CSC IT Center for Science, Finland, for computational resources. Xin is supported by the Academy of Finland for postdoctoral researcher project (grant 331146).

H. Chen, X. Liu, and G. Zhao are with the Center for Machine Vision and Signal Analysis, University of Oulu, FI-90014, Finland. J. Shi is with the School of Software, Xi'an Jiaotong University, China.(E-mail: chen.haoyu, xin.liu, jingang.shi and guoying.zhao@oulu.fi)

Source code is available: https://github.com/mikecheninoulu/THDHMM BiLSTM

Fig. 1, gestures of "waving hands" and "throwing" might share the same beginning phase "raising hands". It is ambiguous and noisy to force systems to achieve correct recognition in the early phase of the gestures. In the previous work, the ambiguous phases of gestures are not considered intensively. Therefore, we introduce *entropy* to measure this kind of uncertainty of recognition, which can be encoded as global temporal information to guide the online HMM decoding.

The segmentation-involved task for online gesture recognition is also a difficult challenge. An extra detection classifier for distinguishing motions and non-motions is always used for enhancing the segmentation in the post-processing [16] [17]. It could not only compensate for the variability of the gestures but also reduce the noise from the non-gesture motions. However, it is desirable to design a system which is advantageous to conduct the segmentation and recognition simultaneously. The conventional HMM has shown the ability to continuously process these temporal semantic connotations. But the transition between the HMM states involves large redundancy and thus it is easy to fall into local optimum as shown in Fig. 1 (a). We improve the conventional HMM by introducing a temporal hierarchical structured dictionary into the HMM decoding. It can not only merge the ambiguous HMM states but also narrow down the search range at every decoding step.

In this work, we propose a DNN-HMM based system for online gesture segmentation and recognition. It consists of three phases: *pre-training*, *training* and *testing* as shown in Fig. 2. In the *pre-training* phase, we model one gesture with sequential HMM states and map the generated HMM states into manifold presentations to measure their distances and relative entropy. Based on the calculated relative entropy, we encode this uncertainty into a temporal hierarchical dictionary (THD). This allows the HMM decoding to tend to the most discriminate HMM states and avoid ambiguity. In the *training* phase, we let the neural network iteratively learn the candidate HMM states by updating the HMM alignment in a progressive manner. In the last *testing* phase, the trained neural network will offer a frame-level prediction and the THD will be used to guide the HMM decoding for the final recognition. Experimental results show that our system achieves better accuracy when compared with the previous work in the online gesture segmentation and recognition tasks.

In summary, we make the following contributions:

1. With the assumption that, the uncertainty of recognizing a gesture only decreases after certain phases are performed, we propose a novel measurement called relative entropy map (REM) and its formulations to investigate the information entropy for discriminate margins of distinct gestural phases.
2. Since forcing neural networks to learn non-discriminate HMM states will bring noise, we propose a progressive learning strategy that the network could learn the HMM states in a more robust manner by iteratively updating the HMM alignment.
3. We achieve start-of-the-art performance for gesture segmentation and early recognition on three well-known datasets: Chalearn 2014, MSRC and OAD dataset. The proposed method shows significant improvements over the previous work on both online and offline gesture recognition tasks.

A preliminary version of this work was presented in [18], but we substantially extend the work in three aspects, which are listed as below: (1) to calculate relative entropy of the gestures, we introduce the manifold presentation with Lie Groups to offer a more robust distance measurement instead of the Euclidean distance; (2) to avoid non-discriminate HMM states, we propose a novel iterative learning algorithm for neural networks to learn a robust recognition of HMM states; (3) we merge the THD into traditional HMM decoding and achieve a further improvement.

The rest of this paper is organized as follows. In Section II, we introduce some related state-of-the-art approaches. In Section III, we give an overview of our online human gesture segmentation and recognition system. In Section IV, we present the gesture motion modeling with HMM and our investigation on the HMM states by measuring their relative entropy. The details of constructing a THD by reducing its relative entropy are also introduced. In Section V, we introduce our hybrid DNN-HMM framework and its progressive training details. Section VI presents the experimental results and discussion, and Section VII concludes the paper.

II. Related work

We provide a brief review of the previous work that is related to our task. Methods for the offline setting and the online setting will be compared separately. Then dictionary-based methods will be discussed.

A. Offline Segmentation and Recognition for Skeleton-based Gestures

One main attribute of the offline setting is that the global temporal information could be utilized for inference. The most famous one might be the improved dense trajectories (iDT) [9]. It utilizes Fisher vectors to describe the motion trajectories as a whole. In the work of Neverova et al. [16], multi-scale networks with distinct steps are used to extract global temporal information of gestures. Yan et al. [19] proposed a spatial-temporal graph convolutional network for skeleton-based recognition. They regarded the skeletal joints as vertexes, and the natural intra-body connections as edges of the graph convolutional networks. Song et al. [20] proposed an end-to-end spatio-temporal attention model for action recognition from skeleton data with LSTM networks. Given the global context, their model could selectively focus on discriminate joints and frames of the inputs. However, the superb performance of most published methods relies on fully observing the whole sequence, which are limited to implemented to the online setting.

B. Online Segmentation and Recognition for Skeleton-based Gesture

For the online recognition task, Ma et al. [21] specified the task as "after observing only a fraction of the activity,

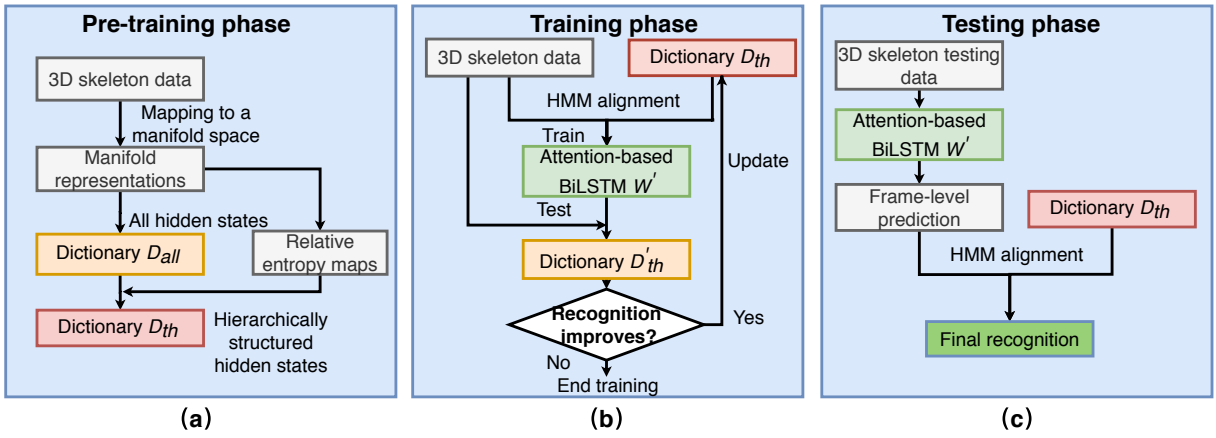

Fig. 2. The overview of the system learning process. (a) In the pre-training phase, we construct the THD Dth with the relative entropy of the candidates. (b) In the training phase, we iteratively train the neural network W with re-aligned HMM states and update HMM alignment with the recognition result predicted from the W. The updated dictionary is obtained Dth when the recognition result stops improving. (c) In the testing phase, given any 3D skeleton data, we use classifier W to obtain a frame-level prediction and align the HMM states with the Dth dictionary to achieve the online recognition.

the classification and duration time of it should be given in an online recognition problem". In this paper, we will measure the performance with the similar idea for online gesture recognition. Considering the particularity of the online model, most approaches always process gestures via sequential temporal steps with local temporal information. To this end, many hand-crafted features are proposed for the local dynamic features. The features include Lie-group features that map skeleton joints to the manifold space to obtain the non-linear properties [22], MovingPose [23] and eigenjoint features [24] that utilize a 3D kinematics descriptor to represent skeleton poses, histograms of 3D joints [7] using histograms to represent high-dimensional skeleton joints, and Cov3D (covariance 3D) features extracted through a spatio-temporal covariance descriptors [25]. As these methods themselves cannot capture the global temporal evolution of gestures, many sequetial temporal modeling methods are further utilized, such as the hidden Markov model (HMM) [18], sliding-window [26] and recurrent neural networks (RNN) [27]. With these temporal modeling framework, the online segmentation is always conducted as an extra gesture class in the sequential recognition.

C. Dictionary-based Methods for Gesture Recognition

Unlike human activities and actions, human gestures are more symbolic and semantic with several distinctive phases. Those phases are spatially and temporally sparse with large redundancy. Thus, dictionary-based learning for the human gesture and action representation has been intensively studied. An early dictionary-based method for gesture recognition is the work of Ivan and Juan [28]. They proposed a spatial hierarchical dictionary by mapping k body parts to higher-level poses with k-means, and several poses are combined for representing the highest action level. But the temporal structures of the body motions are not embodied in these dictionaries. Meanwhile, with the temporal information being considered, the spatio-temporal structured dictionaries show good performances in [29] [30]. The method of [30] is to construct a temporal hierarchical dictionary by specifying the dictionary elements at each temporal steps. However, this method is limited within stationary situations: the whole pre-segmented gestures must be provided to obtain distinct time steps. At last, to sparsify the elements in the dictionary, sparse representation-based methods like K-SVD [31] [32] have been proposed for the gesture recognition task. However, there are few efforts to provide a quantitative measurement of the information redundancy in the dictionary. In this work, as a commonly used tool in the field of information theory, the *entropy* is utilized by us to quantitatively investigate the information redundancy in the dictionary. It can show the information compression capability of a dictionary and encode the global temporal context into it by reducing the entropy of the dictionary.

III. SYSTEM OVERVIEW

The three phases of our system are illustrated as shown in Fig. 2.

In the first *pre-training* phase as shown in Fig. 2 (a), our goal is to obtain a temporal hierarchical structured dictionary, denoted as D_{th}. th stands for the temporal hierarchy for short. We first use the HMM model to generate sequential HMM states of the given gestures, while these HMM states are regarded as the candidates of a dictionary denoted as D_{all}. all stands for the original dictionary with all candidates for short. To sparsify the D_{all}, we first calculate the distances of the candidates in the manifold representation. Then we use these distances to measure the relative entropy in the dictionary D_{all}. Based on this relative entropy, the uncertainty of recognizing those candidates are encoded with a temporal hierarchical structure. At last, the THD D_{th} is sparsified by reducing the relative entropy at each temporal step. It also allows the HMM decoding to yield the most discriminate candidates and avoid ambiguity.

Then, in the next *training* phase as illustrated in Fig. 2 (b), we adopt an attention-based Bi-directional LSTM network (attention-BiLSTM) as a frame-level classifier W'. The output of the classifier W' is enhanced by an HMM alignment to

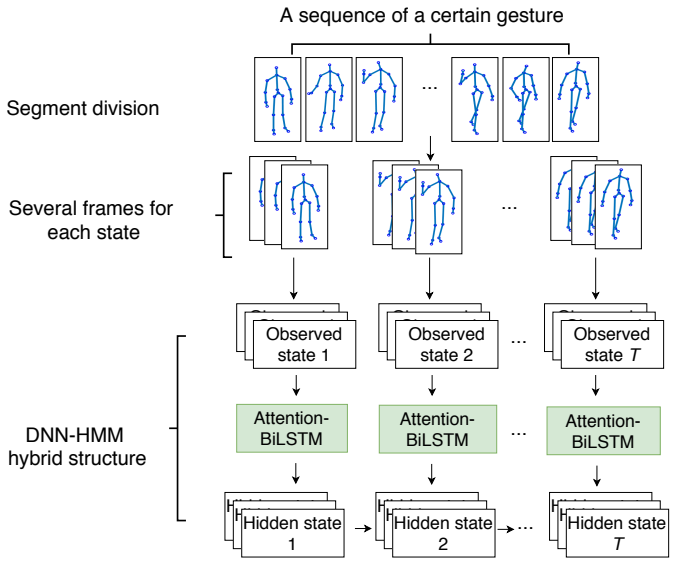

Fig. 3. The abbreviated presentation of modeling gesture motions with the DNN-HMM hybrid structure. The observation layer of the HMM will be the input layer of an attention-BiLSTM network. The output of the attention-BiLSTM is the emission probability of the HMM, which will estimate the hidden states.

produce the recognition result of gestures. Commonly, the training procedure will complete at this phase in other conventional neural network training methods. While in our work, we utilize the generative capability of HMM, to generate a new HMM alignment as the new labels to update the dictionary D_{th} and re-train the classifier W'. The above operation will be conducted iteratively until the recognition accuracy no longer improves. As a result, an enhanced dictionary D_{th}, an updated HMM transition matrix and a robust classifier W' are obtained at this phase.

In the *testing* phase as shown in Fig. 2 (c), given a continuous skeleton sequence, we apply the classifier W' to estimate the HMM states at frame level and HMM alignment is then conducted to identify the final categories. Due to the guide of the THD D_{th}, our system can achieve early detection and recognition of gestures from non-stationary skeleton data streams.

IV. CONSTRUCTING DICTIONARY WITH RELATIVE ENTROPY METHODOLOGY

In this section, we will introduce how to model gesture motions with HMM, and measure the relative entropy of the HMM states with their distances in the manifold representation. At last, based on the relative entropy, we will represent how the THD Dth is constructed by reducing the relative entropy.

A. Gesture Motion Modeling with Conventional HMM

The conventional HMM is one of the most common generative models for modeling time series of observations. As shown in Fig. 3, we model the gesture motion procedures with HMM using similar techniques as [18]. Specifically, given C gesture categories, let $C_{all} = \{1, ..., C\}$ be the set of the given gestures categories with their lengths of arbitrary frames. Then for each gesture, we segment its sequence averagely into T temporal segments to model it as a set of sequential connotations. Therefore, for each gesture class c, a set of temporal segments, or say sub-gestures, $G(c) = \{s_{(c)}1, ..., s_{(c)}T\}$ is defined. The term $s_{(c)}t$ stands for the t^{th} segment of gesture c. Each temporal segment will be mapped to an HMM state and the temporal modeling is achieved by the transition between those HMM states.

As illustrated above, we initially defined the frame boundaries of the HMM states by averagely dividing the gestures. The boundaries of those HMM states will be revised and become more appropriate with progressive learning. We will discuss the learning details in Section V.

Based on the conventional HMM, the full probability of HMM during the training phase is specified as:

$$p(x_1, x_2, ..., x_T, h_1, h_2, ..., h_T) = p(h_1)p(x_1|h_1)\prod_{t=2}^{T} p(x_t|h_t)p(h_t|h_{t-1}), \quad (1)$$

where $p(h_1)$ is the prior probability, $p(x_t|h_t)$ is the observation probability, or known as, the emission probability and $p(h_t|h_{t-1})$ is the transition matrix. For the visible layer in HMM, we denote input skeletal features of the current state as its observed variables x_t to this observed state X_t. The observed variables here are equal to the feature vectors $x_t \in \mathbb{R}^D$ from a D-dimensional input space representing the skeletal gesture for $t = 1, \cdots, T$. One observed state X_t for $t = 1, \cdots, T$ will be mapped to a corresponding hidden state H_t for $t = 1, \cdots, T$ with an attention-based BiLSTM network. The hidden variables of the hidden states are denoted as h_t in the hidden layer of HMM. For a certain gesture c, its hidden state h_t stands for its sub-gesture $s(c)t$.

The attention-BiLSTM network will provide the emission probability as:

$$p(x_t|h_t) = \frac{p(h_t|x_t)p(x_t)}{p(h_t)}, \quad (2)$$

where $p(h_t|x_t)$ is the HMM state posterior probability estimated by the attention-BiLSTM model and $p(h_t)$ is the prior probability of each hidden state. And $p(x_t)$ is the prior probability of each observed state.

By collecting all the HMM states of the given gestures as candidates, we can build the initial dictionary as $Dall = [G(1), ..., G(C)]$ for the HMM transition, and the total number of the candidates in it is $C \times T$.

B. Generating Relative Entropy with Manifold Presentation

1) Relative Entropy : The intuition in this work is that, in the online recognition, the early phases of different gestures present the similarity with high uncertainty of recognition. The gestures only become discriminate after certain phases being performed. Thus, we introduce *entropy* to measure this kind of information chaos level at each distinct temporal step. The *entropy* is always used to measures the disorder degree or randomness within a given system. As it is often used in data compression and encoding information, here *entropy* can also

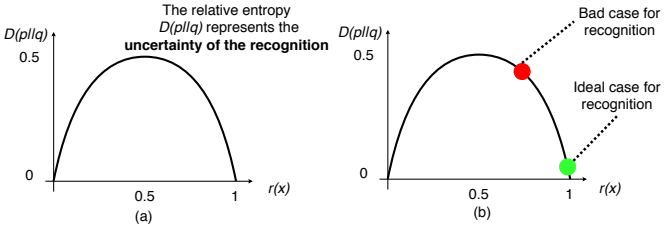

Fig. 4. The abbreviated presentation of the relationship between the mass function $r(x)$ and the relative entropy $D(p||q)$. To make it easier to understand, the mass function $r(x)$ can be regarded as the confidence of recognizing two HMM states p and q. (a) The curve shows the relationship of the distance between two HMM states and the relative entropy $D(p||q)$. (b) Two example cases in the recognition. The green point stands for an ideal case that the value of relative entropy $D(p||q)$ is low for the large confidence of distinguishing the two HMM state. The red point stands for a bad case that the value of relative entropy $D(p||q)$ is high which is caused by the uncertainty of recognizing two HMM states.

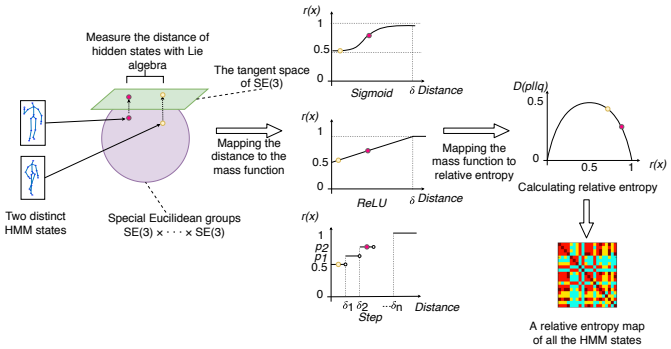

Fig. 5. An illustration of the calculation of mass functions. On the left, it shows that the 3D skeleton data of two hidden HMM states is mapped to Special Euclidean Groups $SE(3) \times ... \times SE(3)$ to obtain a manifold presentation. The distance between the hidden HMM states is then calculated by their Lie algebra $\mathfrak{se}(3)$ on the tangent space. Then on the right, it shows the projective functions proposed by us to map a distance of manifold space to a mass function.

reveal the ultimate compression of the HMM states in the given original dictionary $Dall$. Our goal is to compress the dictionary by reducing the relative entropy at each time step and obtain its temporal hierarchy.

Specifically, here we regard each HMM state in the dictionary as a chaotic system [33]. Obviously, one HMM state by itself is isolated and immeasurable from the perspective of statistical probability. Thus, we introduce *relative entropy* to compare two HMM states as [33]:

$$D(p||q) = \sum_{x} p(x) log \frac{p(x)}{q(x)}, \tag{3}$$

where $p(x)$ and $q(x)$ are called probability mass function in relation to two HMM states needed to be compared. The relative entropy $D(p||q)$ is measured by $bits$, as a measurement of the uncertainty level of distinguishing two probability mass functions $p(x)$ and $q(x)$ from each other.

However, as $p(x)$ and $q(x)$ are discrete distributions, directly calculating the relative entropy of them will encounter infinity value issue. Thus, we introduce an approximate function $r(x)$ to jointly estimate the difference of the function distributions. It can be regarded as a specific term to measure the distance of the two HMM states. Then, instead of measuring the distributions of q and p separately, we can directly calculate the entropy $H(r)$ of $r(x)$ to obtain $D(p||q)$.

$$D(p||q) \sim H(r). \tag{4}$$

A further detailed explanation of relative entropy is elaborated as shown in Fig. 4. The relationship of the relative entropy $D(p||q)$ and mass function $r(x)$ are demonstrated as curve in Fig. 4 (a). In order to make the analysis straightforward, we map the distance of two HMM state to the interval $0.5 < r(x) < 1$. Then, when two HMM states are similar, they will have a high relative entropy, or say the uncertainty of recognition.

The curve presents some basic properties of the relative entropy. When $r(x)$ closes to 1, it means the distance between the two HMM states is large and the confidence of distinguishing the HMM states is high. Thus, the relative entropy has a low value indicating little uncertainty, which is an ideal case shown as the green point in Fig. 4 (b). The bad case for recognition is shown as the red point in Fig. 4 (b). With the distance of the HMM states being relative low, the value of $D(p||q)$ is high because of the ambiguity of the two HMM states. It means the two HMM states are not discriminate enough to contribute the recognition which should be merged.

2) Distances of HMM States in Manifold Presentation: In order to apply relative entropy for gesture case, we need to construct the mass functions $r(x)$ in Eq. 3 for the given two HMM states. Since the mass function $r(x)$ is strongly related to the geometrical distance between the two HMM states, we can set it based on their geometrical distance.

Intuitively, we can simply use Euclidean distance of the 3D skeleton space coordinates to obtain geometrical distance. But for human skeleton data, the Euclidean distance only serves as a similarity measure and can not offer a reasonable distance metric in high dimensional space. We seek a representation that is highly discriminate in high dimensional space.

Instead of using the Euclidean distance, we adopt the work of [22] that uses manifold representation for 3D skeletal data into our relative entropy definition. Mapping 3D skeletal data to manifold presentations as Lie groups will offer a true distance metric and avoid the problem of the *curse of dimension*. Details could be found in [34] and [35] for a general introduction to Lie groups and Special Euclidean Group $SE(3)$.

The special Euclidean group, denoted by $SE(3)$, is also known as a Lie group with the set of all 4 by 4 matrices of the form:

$$P(R; \vec{d}) = \begin{bmatrix} R & \vec{d} \\ 0 & 1 \end{bmatrix}, \tag{5}$$

where $\vec{d} \in \mathbb{R}^3$ denotes the translation vector, and $R \in \mathbb{R}^{3\times3}$ is a rotation matrix.

Precisely, given two rigid body parts at a HMM state, their relative geometry can be described by the rotation and translation. Then we can use the rotation and translation representation of the body pairs to map the 3D skeleton to the Lie groups (as shown in Fig. 5). Thus, the relative geometry

between two body pairs m and n at time instance t can be described using:

$$P_{m,n}(t) = \begin{bmatrix} R_{m,n} & \vec{d}_{m,n} \\ 0 & 1 \end{bmatrix} \in SE(3), \tag{6}$$

$$P_{n,m}(t) = \begin{bmatrix} R_{n,m} & \vec{d}_{n,m} \\ 0 & 1 \end{bmatrix} \in SE(3). \tag{7}$$

However, the Lie group $SE(3) \times ... \times SE(3)$ is a curved manifold and the measurement of the distance in this space is not a trivial task. To tackle this issue, it is common to map the Lie group presentations of the HMM states from $SE(3) \times ... \times SE(3)$ to their Lie algebra $\mathfrak{se}(3)$. The Lie algebra of $SE(3)$ is defined as the tangent plane to $SE(3)$ at the identity element $I4$. It is a 6-dimensional vector space formed by a set of all 4 by 4 matrices of the form $\begin{bmatrix} U & \vec{w} \\ 0 & 0 \end{bmatrix}$, where $\vec{w} \in \mathbb{R}^3$ and U is a 3 by 3 skew-symmetric matrix. For each element in the manifold curve, it is presented with form:

$$B = \begin{bmatrix} U & \vec{w} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -u_3 & u_2 & w_1 \\ u_3 & 0 & -u_1 & w_2 \\ -u_2 & u_1 & 0 & w_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in \mathfrak{se}(3). \tag{8}$$

Then, we can use the exponential map exp: $\mathfrak{se}(3) \rightarrow SE(3)$ and the logarithm map $log_{SE(3)}$: $SE(3) \rightarrow \mathfrak{se}(3)$ between the Lie algebra $\mathfrak{se}(3)$ and the Lie group $SE(3)$ to measure the distance of two HMM states, which are given by

$$exp_{SE(3)}(B) = e^B, \tag{9}$$

$$log_{SE(3)}(P) = log(P), \tag{10}$$

where e and log denote the usual matrix exponential and logarithm respectively.

Thus, for each HMM state of gesture c among the T different temporal steps, it can be represented as $S_c(t) = [P_{1,2(t)}, P_{2,1(t)}, ..., P_{M1,M(t)}, P_{M,M1(t)}] \in SE(3) \times ... \times SE(3)$, where M is the number of body parts.

Using this kind of manifold representation, we measure the distances of two distinct HMM states of two gesture classes c_1 and c_2 with l^2 norm at the same temporal step as:

$$L_{p,q} = \|S_{c_1}(t) - S_{c_2}(t)\|^2, \tag{11}$$

where $L_{p,q}$ can be used to form their relative entropy in the next part.

3) From Manifold Distance to Relative Entropy Map: After obtain the geometrical distance $L_{p,q}$ of two HMM states, we can construct the approximation function $r(x)$ for Eq. 4

Let $r(x)$ be the mass function of approximation distribution of the mutual relationship between p and q with manifold representation $S_c(t)$ of gestures c_1 and c_2 at temporal step t. Then, we further define the projective function f_{proj} to map $L_{p,q}$ into distribution $r(x)$:

$$r(x) = f_{proj}(L_{p,q}(x)). \tag{12}$$

As shown in Fig. 5, we chose three commonly used projective functions as $sigmoid$, $ReLU$ and $step$ to perform the projection $f_{proj}()$.

Algorithm 1 Hierarchical clustering for THD

Input: σ: the threshold relative entropy
T: the number of temporal states in a HMM
C: the number of gesture categories
$Dall(C,T)$: A dictionary of T temporal states, and each temporal state contains C kinds of HMM states from the C gestures at that state.

Output: $Dth(n,T)$: The THD of T temporal states, and each temporal state contains n clusters, the n changes in each temporal states.

```
for t in T do
  calculate REM using Eq.3
  if t = 1 then
    clusters ← Dall(C, 1)
  else
    clusters ← Dth(n, t − 1)
  end if
  unclustered ← clusters
  i = 1
  for cluster in clusters do
    while not (unclustered is null) do
      select baseline (the state with min relative entropy to the rest).
      for state in unclustered do
        if REM(state, baseline) > σ then
          merge state to baseline as Dth(i, t)
          delete state from unclustered
        end if
      end for
      i ++
    end while
  end for
end for
```

The $sigmoid$ projective function is constructed as f_{proj1} :

$$f_{proj1}(L_{p,q}) = \frac{0.5}{1 + e^{-(L_{p,q} - \delta)}} + 0.5, L_{p,q} \geqslant 0, \tag{13}$$

where δ is the threshold to fit the function to represent the projection and set as half of the maximum distance, $L_{p,q}$ is the distance calculated from Lie algebra (the same as below).

The $ReLU$ projective function is built as f_{proj2}:

$$f_{proj2}(L_{p,q}) = min(1, \theta L_{p,q} + 0.5), L_{p,q} \geqslant 0, \tag{14}$$

where θ is coefficient of the $ReLU$ function and assigned with the inverse of the value of the maximum distance.

The $step$ function is formed as f_{proj3}:

$$f_{proj3}(L_{p,q}) = \begin{cases} 1 & 0 \leqslant L_{p,q} \leqslant \delta_1, \\ p_1 & \delta_1 \leqslant L_{p,q} \leqslant \delta_2, \\ ... & \\ p_{n-1} & \delta_{n-1} \leqslant L_{p,q} \leqslant \delta_n, \\ 0 & L_{p,q} \geqslant \delta_n, \end{cases} \quad 0 \leqslant p_{n-1}...p_1 \leqslant 1, \tag{15}$$

where $\delta_1, ..., \delta_N$ are the thresholds for each stage of the step function.

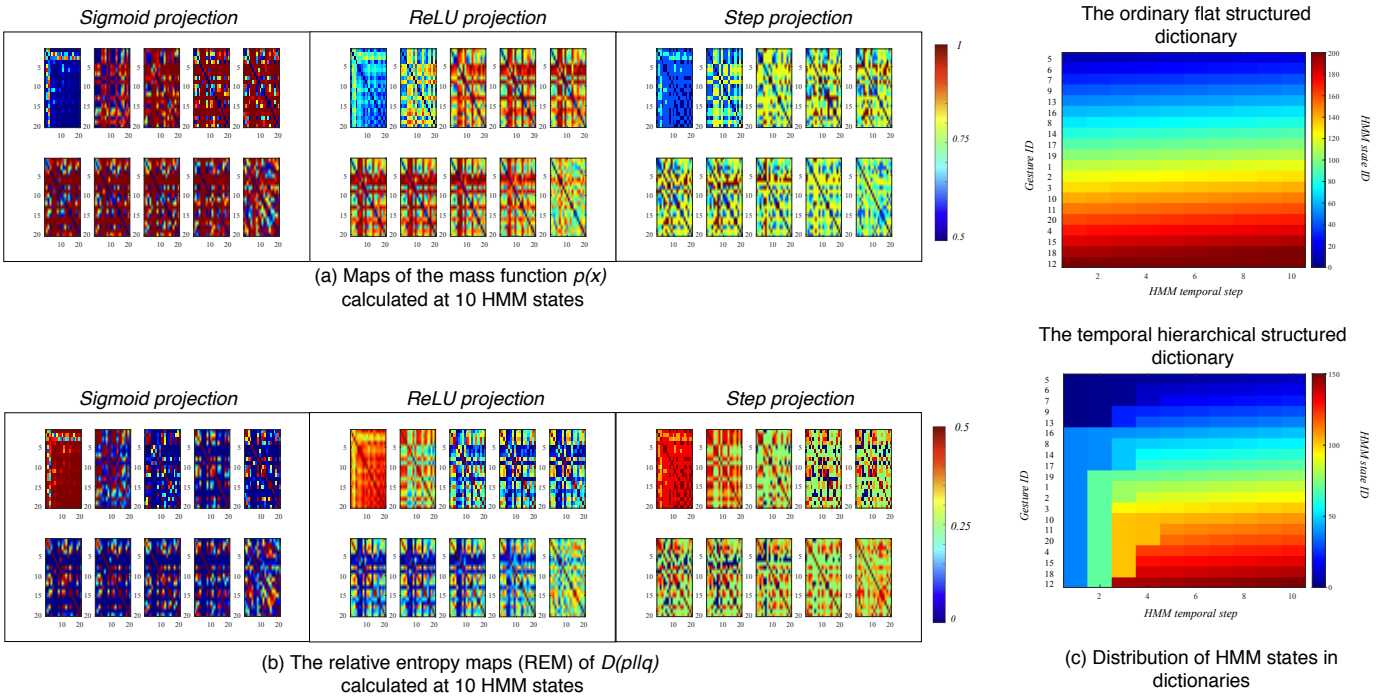

Fig. 6. The mass function and relative entropy maps calculated based on the Chalearn 2014 dataset. The top part (a) shows the three distinct mass function $p(x)$ maps calculated from the Lie algebra distances with the three different projective functions: f_{proj1}, f_{proj2} and f_{proj3}. For each mass function $p(x)$ map, there are ten sub-maps which are in relation to the ten temporal steps. Since the dataset contains 20 gesture categories, each mass function map has 20 columns and 20 rows in which all the gestures classes are compared. Similarly, the bottom part (b) shows the three distinct relative entropy $D(p||q)$ maps calculated from the mass functions of those three projective functions. The cross point at column m and row n in the t^{th} map stands for the relative entropy of gesture m and gesture n at the same time step t. The right part (c) shows the distributions of HMM states in the two kinds of the dictionary. The top one is an original flat structured dictionary and the bottom one is a THD.

Then, at each temporal step of the HMM, we can calculate the relative entropy of the HMM states of their corresponding gesture classes. The implemented the relative entropy calculation for skeletal gesture recognition task is formulated as below:

$$\begin{aligned} D(p||q) \sim H(r) &= -\sum_{x} r(x) log\, r(x) \\ &= -\frac{\sum_{n=1}^{N} \sum_{x} f_{proj}(L_{p,q}(x)) log f_{proj}(L_{p,q}(x))}{N}, \end{aligned} \tag{16}$$

where n stands for the training gesture sample index for $i = 1, \cdots, N$.

At last, the relative entropy map (REM) can be calculated based on the Eq. 16 as:

$$REM = \{D(p_{m,t}||q_{n,t}) \mid t = 1, \cdots, N\}, \tag{17}$$

where $m, n \in C_{all}$ stand for the gesture index. We then investigate the HMM states from all the gesture classes with REM. The result is shown in Fig. 6. The parameter settings for the different projective functions can be seen in the later experimental section.

According to Fig. 6, we can see that, at the beginning steps, the overall entropy is much higher than that of the following temporal steps and thus can be largely compressed. Taking the entropy information calculated above into account, we conclude the below criteria for constructing a THD:

1. Gesture configurations registered at hidden layers are non-repetitive movements and its temporal stream is irreversible.
2. With the information captured from the gesture accumulating, the confidence of recognizing a gesture will keep increasing based on the past information captured.
3. For the gestures at a certain temporal step, the uncertainty of distinguishing them is proportional to the geometric distance of their high dimensional representations, which can be measured by relative entropy. It also reveals the information randomness at that stage.

4) Temporal Hierarchical Dictionary (THD): In the light of the criteria above, we exploit the proposed REM to construct a THD, with which the candidate HMM states can be merged and organized with a temporal structure.

Specifically, we propose a hierarchical clustering algorithm to build the THD. The clustering process is conducted as, at each temporal step, the HMM states from all the gestures at that temporal step are collected as candidates. Then, we cluster those candidates by setting a minimum relative entropy as the threshold. The clustering details can be seen in Algorithm 1.

This will result in a hierarchical structure in the temporal direction. For instance, due to the high relative entropy at the beginning step, few HMM states will be assigned at this time step to reduce the uncertainty of recognition. When it comes to the latter states with the entropy decreasing, more HMM states will be assigned at that temporal step for more confidence in the exact gesture recognition.

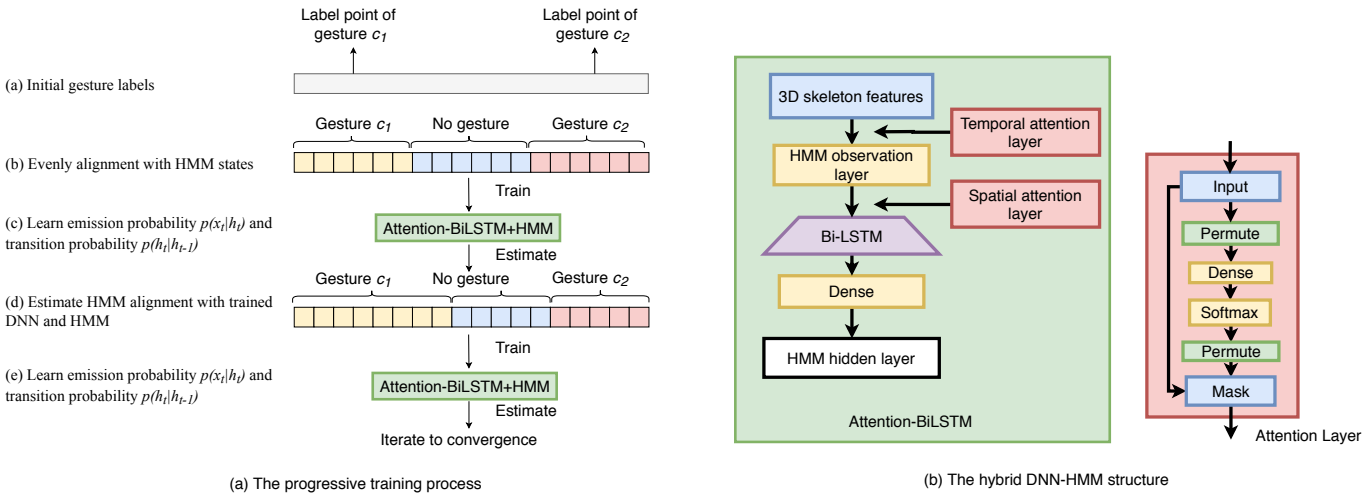

Fig. 7. Network architecture for our hybrid DNN-HMM model. The temporal attention layer computes frame-agnostic attention weights for each LSTM step. The spatial attention layer computes feature-agnostic attention weights for each GRU. The final output is a probability estimation of HMM hidden states.

Here are several principles to implement this automatic clustering. First, the clustering procedure is unidirectional and irreversible. Second, the minimum relative entropy always reaches before mid states, as shown in Fig. 6 (a) (b). It means the distinction of the gestures is done before the ending and the rest HMM states will be uncompressed and all kept. At last, too small relative entropy will cause a severe information loss. Thus, the threshold of the minimum relative entropy should be set carefully, which can be adjusted automatically in future work.

The final constructed temporal hierarchical structure has several advantages over the original one: (1) instead of traversing all the HMM states in the dictionary at each decoding step, it narrows down the search range and largely reduces the number of HMM states used in the transition phase; (2) it embeds temporal hierarchical information into the decoding for a more discriminate state to make a recognition; (3) it utilizes the relative entropy to capture the information change of a temporal sequence, which produces a dictionary with less redundancy. The distributions of HMM states in both the THD and the original dictionary are presented in Fig. 6 (c). We can see that the HMM states in a THD are much more sparse to avoid redundancy and ambiguity.

V. PROGRESSIVE LEARNING OF HYBRID DNN-HMM SYSTEM

In this section, we introduce an iterative re-alignment strategy to train the hybrid DNN-HMM framework as shown in Fig. 7.

Firstly, we define the classifier W' for the HMM state recognition as shown in Fig. 7 (b). We exploit a Bi-LSTM network as the classifier W' for the HMM hidden state estimation. The propagation of the Gated Recurrent Units (GRU) in the Bi-LSTM layer is both forward and backward. The input of the Bi-LSTM network is a five-step 3D skeleton feature extracted from the current five frames. Skeleton features from each frame are fed into each step of the LSTM. Besides, we introduce the attention mechanism into the network for both spatial and temporal domains. A temporal attention layer is used for computing frame-agnostic attention weights for every LSTM step, and a spatial one is used for that of each GRU. At last, the final structure of the classifier W' is a Spatial-Temporal Attention BiLSTM network, called STABNet for short.

Training the system is in a progressive procedure for both the HMM and the classifier W' (the STABNet). The whole training process is elaborated as shown in Fig. 7 (a). In the initialization, one gesture sequence is divided evenly into T temporal segments as T HMM states. The STABNet is then trained with the initial alignment as ground truth. During the training phase, Viterbi algorithm is used to give the utmost HMM alignment among all the possible paths efficiently [36]. The decoded sequence $\hat{g}$ is determined as:

$$\hat{g} = arg \max_{p} p(x_1, x_2 \dots x_T, h_1, h_2 \dots h_T), \tag{18}$$

where x_t and h_t are the observation state and hidden state of a gesture at temporal step t. To obtain the maximum value for this emission probability, we get:

$$\begin{aligned} &p(x_1, x_2 \dots x_T, h_1, h_2 \dots h_T) \cong \\ &\pi(h_0) \prod_{t=2}^{T} p(h_t|h_{t-1}) \prod_{t=1}^{T} p(x_t|h_t). \end{aligned} \tag{19}$$

By using Eq. 19, we can break down the problem of solving the utmost probability of main gesture class into solving HMM states probability with hidden states $h_1, h_2, \dots, h_T$. Once the HMM states being aligned, the test sequence can be inferred with gestures being segmented and recognized.

After the STABNet and HMM being trained with the initial HMM alignment, they can be used to estimate a new HMM alignment. The new alignment can be used as labels to retrain the framework with Eq. 18. The new aligned HMM path can be more accurate and natural than the previous one. This step will be iterated until convergence as shown in Fig 7 (a).

With the framework iteratively approximates the optimal HMM hidden states on the training data, we define a stop

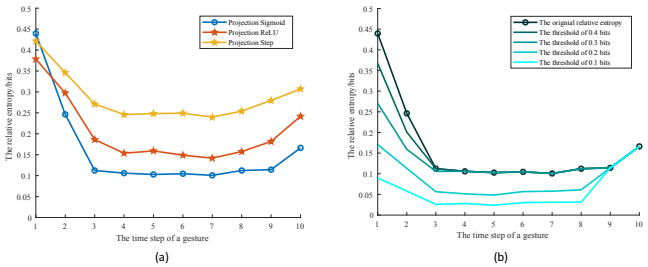

Fig. 8. (a) A comparison of using different three projective functions. The sigmoid one shows a better representing capability than that of the ReLU function and step function, especially in the beginning temporal steps. (b) A comparison of the relative entropy of the original dictionary and the four distinct THDs. The original dictionary contains the highest relative entropy at each time step. The four compared THDs are obtained by clustering the ambiguous HMM states to reduce the relative entropy. The thresholds of the min relative entropy are set as 0.4, 0.3, 0.2 and 0.1 bits respectively.

criterion based on the training accuracy. Let $valacc(i)$ denote the validation accuracy of the training set at iteration i. We stop the iteration if the accuracy improvement is less than a threshold θ,

$$valacc(i) - valacc(i-1) < \theta \rightarrow stop. \tag{20}$$

In practical training, to ensure the new alignment is correct, before the new HMM alignment, we enforce the start three and the end three frames to be labeled as the pre-designed HMM states in the dictionary. Also, we set a minimum loss to avoid over-fitting. An early stop will be conducted if the minimum loss is achieved or the improvement of the validation accuracy is less than θ.

To realize the online segmentation task, we introduce an extra non-gesture class as a specific HMM state. The segmentation is conducted by decoding the HMM states with the transition of the non-gesture class and gesture classes.

VI. EXPERIMENTS

In this section, details of the experiments are illustrated. We evaluate our method on three well-known gesture datasets: Chalearn 2014 gesture dataset [37], MSRC dataset [38] and online action detection (OAD) dataset [39]. In all these datasets, multiple gesture or action instances are contained in each long skeleton stream, thus the segmentation task is involved.

A. Datasets

ChaLearn 2014 dataset includes 20 Italian sign gesture categories with 20 joints estimated by Kinect in the dataset. It contains 470 labeled sequences for training, 230 sequences for validating and the rest 240 sequences for testing. In each sequence, 10 to 20 gestures are performed and in total, more than 14,000 gestures are contained in the whole dataset.

MSRC gesture dataset includes 594 sequences from 30 subjects performing 12 kinds of gestures with 20 joints estimated by Kinect. Note that gestures in the MSRC dataset are only labeled at the mid point of each gesture motion instead of a certain boundary of the gesture duration. To train the network, we encode a window of 90 frames centered on the gesture point from the ground truth. This coarse labeling can be improved with the later progressive learning.

The last OAD action dataset includes 10 daily human action categories. It was captured as long video sequences with Kinect v2. The start and end frames are annotated. There are more than 50 long sequences in total and 30 of them are used for training, 20 for testing and rest of the sequences are for processing speed validation.

B. Features

In the Charlearn 2014 and MSRC datasets, for the computational-consuming issue and gesture-orientated goal, we only use the coordinates of the 11 skeleton joints from the upper body. Then we adopt the work of [23] and extract features called "MovingPose". It is an efficient feature for skeleton data including relative position, relative velocities/accelerates and joint angles, which results in a feature dimension of 154 for each input step of the STABNet. For the skeleton feature extraction, we conduct a normalization to the coordinates. The coordinates are transformed into a person-centric coordinate system with the hip center as the origin.

In OAD action dataset, all the 25 skeletal body joints are used to generate features. "MovingPose" is also adopted to enhance the performance, which results in a feature dimension of 525 for each frame. Since the annotation of OAD dataset is different from others that, the start frame and end frame are within peak duration (not a from-none-to-action pattern), we compensate λ frames to the beginning of actions to learn pre-action information for better online recognition. The λ is set as 12 in the implementation.

C. Building THD with the Relative Entropy

Firstly, we investigate the relative entropy in an ordinary HMM dictionary $Dall$ with REM in Fig. 6. As shown in the figure that, at the beginning steps, the overall entropy is much higher than that of the following temporal steps. It means achieving an exact recognition at starting temporal steps suffers from the high uncertainty of recognition. Only after certain temporal steps being performed, the uncertainty of recognition gets low. Thus forcing an exact recognition before entropy gets lower are not necessary and will even bring noisy.

Then we try to construct the REMs of the original dictionary D_{all} with the three different projective functions. The parameter settings are $\delta = 15$ for f_{proj1}, $\theta = \frac{1}{30}$ for f_{proj2} and $\delta_1 = 4, \delta_2 = 9, \delta_3 = 12, \delta_4 = 20$ for f_{proj3} respectively. The relative entropy calculated with the three projective functions f_{proj1}, f_{proj2} and f_{proj3} for the original dictionary can be seen in Fig. 8 (a). The setting can be slightly different with the sample numbers and datasets.

Obviously, the $sigmoid$ f_{proj1} shows a better representing capability than that of the $ReLU$ function f_{proj2} and $step$ function f_{proj3} especially in the beginning temporal steps. Specifically, the representing capability of the $ReLU$ projective function f_{proj2} (0.22 bits from 0.16 to 0.38) and the $step$ projective function f_{proj3} (0.18 bits from 0.25 to 0.43) are not as good as the $sigmoid$ function f_{proj1} (0.33 bits from

0.11 to 0.44). Thus, we choose the $sigmoid$ function as the projection f_{proj}.

At last, as shown in Fig. 8 (b), using Algorithm 1, we build the THD with the $sigmoid$ projection. In the figure, we quantitatively represent how the entropy changes when the original dictionary D_{all} is temporal hierarchical structured into THD D_{th}. We construct four distinct THDs by setting four distinct thresholds of the minimum relative entropy as 0.4, 0.3, 0.2 and 0.1 bits respectively and represent their the entropy curves. It can be seen that, without processing the ambiguous HMM states, the original dictionary contains the highest relative entropy at each time step. By setting a threshold for ambigous HMM states, like 0.3 and 0.4 bits, all of the relative entropy is reduced and the information is kept. When the threshold is set too low such as 0.1 and 0.2 bits, even though the relative entropy is much lower than the original one, it can cause an excessive compression of HMM states and the information lost. That's why the performance of THD sometimes can be worse than other compared methods, such as lines 1 and 4 in Table II (0.778 of HMM and 0.763 of THD-HMM). Thus, a proper setting of the relative entropy threshold is vital to a THD. In the practical experiments, we fix the HMM state number as 10 and set the thresholds of the minimum relative entropy for the Chalearn 2014 dataset, MSRC and OAD dataset as 0.24, 0.31 and 0.6 bits to obtain the best THDs. Furthermore, we visualize the distributions of HMM states in the original dictionary D_{all} and the produced THD D_{th} are shown in Fig. 6 (b). The ambiguous states are clustered based on the REM in the THD to avoid uncertain recognition.

D. Comparison of Proposed Neural Network with Baseline Architectures

We conduct experiments with the following architectures on Chalearn dataset (parameter setting is slightly different in OAD and MSRC):

(1) STABNet. The spatio-temporal attention BiLSTM network is our proposed model for 3D gesture prediction. We set our seven-layer STABNet architecture as $[N_X,\ N_{tem},\ N_{LSTM1},\ N_{spa},\ N_{LSTM1},\ N_{Dense},\ N_{output}]$. Here N_X is input feature layer, namely, the HMM observation layer with five steps of the LSTM. N_{tem} is the temporal attention layer with five steps. The BiLSTM layers N_{LSTM1} and N_{LSTM2} contain 2000 GRUs and 1000 GRUs separately. The layer between the two BiLSTM layers N_{spa} is the spatial attention layer with 2000 units. The dense layer N_{Dense} of 1000 units is stacked with the sigmoid activation. N_{output} is the total hidden state number.

(2) TABLNet. Temporal-Attention BiLSTM Network (TABNet) is similar to STABNet, but the spatial attention layer is removed. We configure the structure to verify the contribution brought by the temporal attention layer.

(3) SABLNet. Spatial-Attention BiLSTM Network (SABNet) is similar to TABNet, but with the spatial attention layer.

(4) BLNet. It is a basic BiLSTM Network similar to STABNet but without any attention layers.

(5) 2-FC. A two-layer fully connected network is set to verify how the BiLSTM contribute to the recognition result replacing BiLSTM layers with dense connected layers with a 0.1 dropout rate. Note that we just flat the 5-step length features of the LSTM input into a 1-dimensional feature as the input here.

TABLE I
A COMPARISON OF DISTINCT NETWORKS ON CHALEARN 2014 DATASET.

Network structure	Jacc. Score		
	HMM temporal step		
	6	10	14
2-FC	0.756	0.757	0.693
BLNet	0.730	0.732	0.670
TABNet	0.728	0.733	0.701
SABNet	0.770	0.778	0.752
STABNet	**0.790**	**0.812**	**0.761**

TABLE II
A COMPARISON OF THD D_{th} AND ORDINARY DICTIONARY D_{all} ON CHALEARN 2014 DATASET.

	Temporal steps	HMM states	Jacc. Score	Decoding speed(fps)
Oridinary dictioanry	6	120	0.778	646.9
	10	200	0.780	314.3
	14	280	0.754	185.7
THD	6	87	**0.763**	920.4
	10	121	**0.812**	652.1
	14	152	**0.781**	540.4

We present the performance validations of the distinct networks on ChaLearn 2014 dataset as shown in Table I. Note that to make a fair comparison, we compare all the networks with the same skeleton features for the experimental settings. Results in bold for our methods, with underlines for the best method, the same as below.

The experimental results show that, for distinct temporal steps (in 6, 10, 14), a simple fully-connected network can achieve a better recognition result than that of BLNet. The training process can explain it as, a simple fully-connected network can learn the HMM states in a fast speed than a more complex BLNet. As well, the complex BLNet is also hard to be trained. However, by introducing temporal and spatial attention layers into BLNet, the STABNet can gain an improvement of 0.48 in Jacc. Score. It proves that the attention layers could efficiently mine the potential of BLNet. On the other side, a large redundancy (too many HMM states) also causes a significant negative impact on the accuracy which can be seen in the 14-state for all the networks. It is interesting to see that, our STABNet can handle the dictionary with a large size (280 HMM states) and redundancy (0.754 of STABNet to 0.693 to 2-FC) when the temporal step is 14.

E. Comparison of THD and Ordinary Dictionary

A comparison of a THD and an ordinary dictionary is presented in Table II. The results show our THD generally yields better performance and leads to a substantial decrease in the number of HMM states than the original one. When modeling gestures with ten temporal steps, our THD obtains 0.812 Jacc. Score by only using 121 HMM states while an ordinary dictionary can only achieve 0.780 Jacc. Score by

TABLE III
THE PERFORMANCE OF PROGRESSIVE LEARNING ON THE MSRC DATASET.

Iteration	Training accuracy	Validating accuracy	HMM alignment F-score
1	84.1%	69.8%	0.762
2	92.6%	76.8%	0.840
3	95.4%	80.4%	0.867
4	96.2%	81.0%	**0.871**
5	96.9%	80.1%	-

using 200 HMM states. In the case of six HMM states, the performance of our THD decreases and not as good as that of the original one. As demonstrated in Fig. 8, when the relative entropy is already low enough, minimizing it might cause the information loss. In this way, a good THD structure should make a balance between information loss and statistical redundancy.

The HMM decoding speeds of the THD and ordinary dictionary are also compared. For each temporal step setting (6, 10, 14), our THD yields faster decoding speeds than that of the ordinary dictionary (920.4 v.s. 646.9 fps, 652.1 v.s. 314.3 fps, 540.4 v.s. 185.7 fps). The speeds of our system in each setting also prove that the performance of our system is sufficient for online segmentation and recognition tasks.

F. Performance of Progressive learning

We validate the contribution of progressive learning to our system. The performance of progressive learning is presented on MSRC gesture dataset as shown in Table III. Note that in this dataset, the measurement during the training is accuracy, but changes to F-score in the final testing phase.

As shown in Table III, the HMM re-alignment will bring an improvement of 8.5% on the validation set and 0.078 on the test set in F-score. The next re-alignment iteration seems to have a smaller impact with an improvement of 2.8% on the validation set and 0.27 on the test set in F-score. Thus, the initial re-alignment brings significant improvements over the state of the arts. During the iteration, it is prone to get over-fitting for those networks, so we have to use an early stopping strategy to train the networks. If the early stopping strategy did not take part in the training phase, the validating accuracy could be above 98% but the test F-score drops to 0.23, which means the network will be obviously over-fitted without an early stopping.

G. Performance of the Early Detection

Furthermore, the early recognition capability of our framework is validated by using the measurement of Activity Monitoring Operating Characteristic (AMOC) [41]. AMOC is obtained by changing the observational ratio (the proportion of the gesture that has been observed at the time of decision). The evaluation is conducted on both Chalearn 2014 and OAD dataset. Here, we set the observed ratio of the instances ranging from 10% to 100%.

For Chalearn dataset, we compare our THD based methods with several state-of-the-art frameworks for online gesture recognition tasks. As shown in Table IV, generally,

TABLE IV
THE EARLY DETECTION PERFORMANCES ON CHALEARN 2014 DATASET.

	Accuracy				
Observational Ratio	10%	30%	50%	80%	100%
Sliding-LSTM [26]	13.4%	65.1%	64.2%	57.9%	55.6%
Moddrop [16]	20.2%	63.7%	65.2%	66.1%	54.6%
DBN-HMM [40]	57.4%	71.3%	72.0%	72.5%	72.5%
THD-HMM	**28.0%**	**68.3%**	**79.0%**	**80.1%**	**80.1%**
THD-HMM guided	**60.3%**	**73.1%**	**81.2%**	**81.2%**	**81.2%**

TABLE V
A COMPARISON OF THE STATE-OF-THE-ART METHODS ON CHALEARN 2014 GESTURE DATASET.

Method	**Results (Jacc.)**
Fisher Vector [42]	0.747
DNN-ES-HMM [40]	0.787
HOG, Boosted classifier [43]	0.789
HOG, MRF [44]	0.792
ModDrop [16]	0.808
THD-HMM (online)	**0.812**
THD-HMM + postprocessing	**0.834**

HMM-based methods perform better than sliding window-based methods, which is proved by comparing THD-HMM guided (60.3%, 73.1%, 81.2%, 81.2%, 81.2%) and DBN-HMM (57.4%, 71.3%, 72.0%, 72.5%, 72.5%) to sliding-LSTM (13.4%, 65.1%, 64.2%, 57.9%, 55.6%). We offer the performance of only using THD for HMM, it shows better results only after 50% observational ratio (28.0%, 68.3%, 79.0%, 80.1%, 80.1%) than that of the DBN-HMM. It is the cost of reducing the relative entropy that, exact gesture recognition can not be achieved in the beginning several temporal steps as gestures share the same HMM states. So, we use the THD as a prior to guide a HMM decoding with candidates from an ordinary dictionary. It especially works in the beginning temporal steps (from 28.0% to 60.3%). By combining THD into an ordinary dictionary as shown in the last line in the Table, we obtain the best online recognition results at the distinct observational ratios. It performs better (81.2%) than a conventional HMM based method DBN-HMM (72.5%). It is proved that our proposed framework can be implemented to continuous gesture streams and achieve gesture recognition even when they are halfway performed (recognition accuracy of 81.2% at the observational ratios as 50%). Note that, to make it fair and purely compare the recognition performance, all the compared methods above use raw skeleton data as input.

For the OAD dataset, the early detection performance is shown in Table VII. We can see that, our hybrid DNN-HMM framework (STABNet-HMM) can have satisfying performances (above 79.9%, 84.1% and 84.4%) of online recognition at each observational ratio (10%, 50% and 90%). The early recognition accuracy can achieve 87.2% even the actions are only 10% performed.

H. Comparison with the state of the arts

At last, we represent the-state-of-the-art techniques as well as ours on ChaLearn 2014, MSRC and OAD action datasets

TABLE VI
A COMPARISON OF THE STATE-OF-THE-ART METHODS ON MSRC GESTURE DATASET.

Methods	Results (F-score)
Randomized Forestg [38]	0.62±0.04
Structured Streaming Skeletons [45]	0.718±0.159
DBN-ES-HMM [6]	0.7243
THD-HMM	**0.762±0.053**
THD-HMM + progressive learning	**0.871±0.013**

TABLE VII
THE EARLY DETECTION PERFORMANCE OF OUR FRAMEWORK ON THE OAD DATASET USING SKELETON DATA COMPARED WITH THE STOA METHODS.

	Acurracy			Processing speed
Observational Ratio	10%	50%	90%	
ST-LSTM [46]	60.0%	75.3%	77.5%	-
Attention Net [47]	59.0%	75.8%	78.3%	-
JCR-RNN [39]	62.0%	77.3%	78.8%	1230 fps
SSNet [48]	65.6%	79.2%	81.6%	70 fps
STABNet-HMM (RS)	**79.9%**	**84.1%**	**84.4%**	**744 fps**
STABNet-HMM-THD (RS)	**80.9%**	**88.2%**	**88.2%**	**1822 fps**
STABNet-HMM (MP)	**87.2%**	**91.0%**	**91.0%**	**720 fps**
STABNet-HMM-THD (MP)	**87.2%**	**92.0%**	**93.1%**	**1720 fps**

RS: raw skeleton, MP: MovingPose feature.

in Table V, VI and VII respectively.

For the ChaLearn 2014 dataset, we follow the evaluation protocol in [16] and quantify model performance with the Jaccard index (Jacc.). The Jaccard index measures the accuracy of both the classification and the segmentation at the frame level. It is defined as follows:

$$J_n = \frac{A_n \bigcap B_n}{A_n \bigcup B_n}, \tag{21}$$

where A_n is the ground truth of gesture n and B_n is the predicted class for the given gesture. A correct recognition will lead the term $A_n \bigcap B_n$ to be 1 while an incorrect one will lead it to be 0. Good segmentation performance is obtained with a large intersection and a small union of A_n and B_n. Our THD-HMM shows the best performance in skeleton-based gesture segmentation and recognition tasks for both the online and offline settings. Similar to [16], for the offline setting, we implement a detection classifier trained with approximately 93% accuracy to recognize motion and non-motion gestures.

In the MSRC gesture dataset, F-score is always used for validation. It is the harmonic mean of *recall* and *precision* in a tolerated latency. On this dataset, the latency is set as $333ms$, the same leave-subjects-out protocol and settings as [38] and [6] are used. Note that the DBN-ES-HMM method is our baseline method, encoding a conventional HMM dictionary with temporal hierarchical structure will bring around 0.038 improvements in F-score. The progressive learning contributes a significant increase of 0.109. The reason is that MSRC gesture dataset is labeled only with key points of gestures, and this coarse labeling can be fined with progressive learning.

In the OAD dataset, we use the same protocol as [48] that sets different observation ratios to validate the algorithm, thus the accuracy is reported for this dataset. We can see that, our hybrid DNN-HMM framework (STABNet-HMM) can largely improve the performance of online recognition (84.4% with raw skeleton input). By implementing MovingPose, our algorithm gains another substantial improvement (by 6.8% for STABNet-HMM framework). Not only that, our THD can further enhance the performance of STABNet-HMM by 3.8% (with raw skeleton) and 2.1% (with MovingPose features). As a result, our proposed framework THD-HMM with "MovingPose" features performs the best (recognition accuracy of 87.2%, 92.0% and 93.1% at the observational ratios as 10%, 50% and 90%). At last, by comparing STABNet-HMM-THD with STABNet-HMM, it proves that our proposed THD algorithm can largely improve the recognition accuracy and online processing speed by reducing the redundancy HMM states and ambiguity (from 744 fps of STABNet-HMM-RS to 1822 fps of STABNet-HMM-THD-RS).

I. Computational setup

For the Chalearn 2014 dataset, the training parameters are set as 128, 0.01, 0.95, 0.5, 2, 50 for the batch size, the learning rate, the momentum, the factor of reducing learning rate, the early stopping patience and the epoch iteration respectively. For the MSRC gesture dataset, the training parameters are set as 64, 0.01, 0.95, 0.5 for the batch size, the learning rate, the momentum, the factor of reducing the learning rate respectively. We fix the training epoch as 25 to avoid overfitting in this dataset and set the minimum loss of progressive learning as 0.08. In the OAD dataset, the batch size and training epoch number are set as 32 and 10 respectively. The learning rate is fixed as 0.001. The distribution platform is Tensorflow with a single GPU: NVidia 1080Ti (RAM: 12 GB). The CPU is Intel Core i7 8700 with 12 cores.

VII. CONCLUSION

In this paper, for the 3D skeleton based online gesture segmentation and recognition task, we introduce the relative entropy to investigate the redundancy in gesture phases. Based on it, we further propose a Temporal Hierarchical Dictionary with HMM (THD-HMM). The experimental results prove that there exists large redundancy in online HMM decoding process and our THD-HMM can successfully guide the HMM decoding to the most discriminate states and narrowing down the search range of neural networks. THD-HMM can not only largely improve the online and early recognition accuracy but also increase the processing speed. Progressive learning of the attention-BiLSTM is further proposed for a robust recognition of HMM states which is proved efficient in improving HMM alignment at the fine-grained frame level. The experimental results on three gesture datasets show the effectiveness of the proposed method compared with state-of-the-art performances. Future research is to develop the supervised HMM state clustering into unsupervised learning and explore more complementary representations from heterogeneous inputs such as RGB and audio data.

REFERENCES

[1] S. D. Kelly, S. M. Manning, and S. Rodak, "Gesture gives a hand to language and learning: Perspectives from cognitive neuroscience, developmental psychology and education," *Language and Linguistics Compass*, vol. 2, 2008.

[2] J. Shi, I. Alikhani, X. Li, Z. Yu, T. Seppnen, and G. Zhao, "Atrial fibrillation detection from face videos by fusing subtle variations," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[3] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2011.

[4] H. Chen, X. Liu, X. Li, H. Shi, and G. Zhao, "Analyze spontaneous gestures for emotional stress state recognition: A micro-gesture dataset and analysis with deep learning," in *IEEE International Conference on Automatic Face & Gesture Recognition*, 2019.

[5] H. Cheng, L. Yang, and Z. Liu, "Survey on 3d hand gesture recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, Sept 2016.

[6] D. Wu and L. Shao, "Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2014.

[7] L. Xia, C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2012.

[8] Z. Huang, C. Wan, T. Probst, and L. V. Gool, "Deep learning on lie groups for skeleton-based action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.

[9] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *IEEE International Conference on Computer Vision*, Dec 2013.

[10] Y. Zhou, X. Sun, Z.-J. Zha, and W. Zeng, "Mict: Mixed 3d/2d convolutional tube for human action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[11] A. Ben Tanfous, H. Drira, and B. Ben Amor, "Coding kendall's shape trajectories for 3d action recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[12] X. Liu, H. Shi, X. Hong, H. Chen, D. Tao, and G. Zhao, "3d skeletal gesture recognition via hidden states exploration," *IEEE Transactions on Image Processing*, vol. 29, 2020.

[13] W. Peng, X. Hong, H. Chen, and G. Zhao, "Learning graph convolutional network for skeleton-based human action recognition by neural searching." in *the AAAI Conference on Artificial Intelligence*, 2020.

[14] H. Chen, Z. Yu, X. Liu, W. Peng, Y. Lee, and G. Zhao, "2nd place scheme on action recognition track of eccv 2020 vipriors challenges: An efficient optical flow stream guided framework," in *arXiv preprint arXiv:2008.03996*, 2020.

[15] X. Liu, H. Shi, X. Hong, H. Chen, D. Tao, and G. Zhao, "Hidden states exploration for 3d skeleton-based gesture recognition," in *IEEE Winter Conference on Applications of Computer Vision*, 2019.

[16] N. Neverova, C. Wolf, G. Taylor, and F. Nebout, "Moddrop: Adaptive multi-modal gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, Aug 2016.

[17] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 37, no. 3, May 2007.

[18] H. Chen, X. Liu, and G. Zhao, "Temporal hierarchical dictionary with hmm for fast gesture recognition," in *The IEEE Conference on International Conference on Pattern Recognition*, 08 2018.

[19] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[20] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[21] S. Ma, L. Sigal, and S. Sclaroff, "Learning activity progression in lstms for activity detection and early detection," in *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2016.

[22] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014.

[23] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013.

[24] X. Yang and Y. L. Tian, "Eigenjoints-based action recognition using nave-bayes-nearest-neighbor," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2012.

[25] A. Sanin, C. Sanderson, M. T. Harandi, and B. C. Lovell, "Spatio-temporal covariance descriptors for action and gesture recognition," in *Applications of Computer Vision, 2013 IEEE Workshop on*. IEEE, 2013.

[26] I. Lee, D. Kim, S. Kang, and S. Lee, "Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks," in *2017 IEEE International Conference on Computer Vision*, Oct 2017.

[27] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, March 1994.

[28] I. Lillo, J. C. Niebles, and A. Soto, "Sparse composition of body poses and atomic actions for human activity recognition in rgb-d videos," *Image Vision Comput.*, vol. 59, no. C, Mar. 2017.

[29] V. Bettadapura, G. Schindler, T. Ploetz, and I. Essa, "Augmenting bag-of-words: Data-driven discovery of temporal and structural information for activity recognition," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013.

[30] A. Alfaro, D. Mery, and A. Soto, "Human action recognition from inter-temporal dictionaries of key-sequences," in *Image and Video Technology*. Springer Berlin Heidelberg, 2014.

[31] Y. Song, Y. Liu, Q. Gao, X. Gao, F. Nie, and R. Cui, "Euler label consistent k-svd for image classification and action recognition," *Neurocomputing*, vol. 310, 2018.

[32] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent k-svd," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[33] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley and Sons, 2006.

[34] B. C. Hall, "An elementary introduction to groups and representations," *arXiv preprint math-ph/0005032*, 2000.

[35] R. M. Murray, *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[36] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[37] S. Escalera, X. Baró, J. Gonzàlez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. J. Escalante, J. Shotton, and I. Guyon, *ChaLearn Looking at People Challenge 2014: Dataset and Results*. Springer International Publishing, 2015.

[38] S. Fothergill, H. M. Mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012.

[39] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, and J. Liu, "Online human action detection using joint classification-regression recurrent neural networks," *European Conference on Computer Vision*, 2016.

[40] D. Wu, L. Pigou, P. J. Kindermans, N. D. H. Le, L. Shao, J. Dambre, and J. M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, Aug 2016.

[41] M. Hoai and F. De la Torre, "Max-margin early event detectors," *International Journal of Computer Vision*, vol. 107, no. 2, 2014.

[42] G. D. Evangelidis, G. Singh, and R. Horaud, *Continuous Gesture Recognition from Articulated Poses*, L. Agapito, M. M. Bronstein, and C. Rother, Eds. Springer International Publishing, 2015.

[43] C. Monnier, S. German, and A. Ost, *A Multi-scale Boosted Detector for Efficient and Robust Gesture Recognition*, L. Agapito, M. M. Bronstein, and C. Rother, Eds. Springer International Publishing, 2015.

[44] J. Y. Chang, *Nonparametric Gesture Labeling from Multi-modal Data*, L. Agapito, M. M. Bronstein, and C. Rother, Eds. Springer International Publishing, 2015.

[45] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng, "Online human gesture recognition from motion data streams," in *Proceedings of the 21st ACM International Conference on Multimedia*. New York, NY, USA: ACM, 2013.

[46] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal lstm with trust gates for 3d human action recognition," in *European conference on computer vision*. Springer, 2016.

[47] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention lstm networks for 3d action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[48] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. C. Kot, "Ssnet: Scale selection network for online 3d action prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Haoyu Chen received the B.Sc. degree from the China University of Geosciences, Wuhan, China, in 2015, and the M.Sc. degree in computer sciences and engineering from the University of Oulu, Finland, in 2017, where he is currently pursuing the Ph.D. degree in computer science and engineering with Center for Machine Vision and Signal Analysis, under the supervision of Prof. G. Zhao. He was a visiting researcher in Centre for Education and Learning at Delft University of Technology, the Netherlands, from 2019 to 2020. His research interests include action, gesture recognition, and emotional AI.

Xin Liu (Member, IEEE) received the Ph.D. degree in information and communication engineering from Xi'an Jiaotong University, China, in 2016, and the Ph.D. degree in computer science from the University of Oulu, Finland, in 2019. He is currently an Academy of Finland postdoctoral research fellow, and also a Senior Researcher with Center for Machine Vision and Signal Analysis, University of Oulu, Finland. His research interests include human behavior recognition and analysis, emotion understanding, and video background subtraction. He received the IEEE ICME Best Paper Award in 2017. He has authored or co-authored more than 30 papers in prominent journals and conferences, and has served for prestigious conferences and journals, including the IEEE CVPR, ICCV, T-PAMI, T-IP, T-CSVT, T-NNLS, T-CI, IJCV, ACM TOMM and PR

Jingang Shi received the B.S. and Ph.D. degrees from the Department of Electronics and Information Engineering, Xi'an Jiaotong University, China. From 2017 to 2020, he was a post-doctoral researcher with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. Since 2020, he has been an associate professor at the School of Software, Xi'an Jiaotong University. His current research interests mainly include image restoration, face analysis, and biomedical signal processing.

Guoying Zhao (IEEE Senior member 2012) is a Professor with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland, where she has been a senior researcher since 2005 and an Associate Professor since 2014. She received the Ph.D. degree in computer science from the Chinese Academy of Sciences, Beijing, China, in 2005. She has authored or co-authored more than 230 papers in journals and conferences. Her papers have currently over 13100 citations in Google Scholar (h-index 53). She is co-program chair for ACM International Conference on Multimodal Interaction (ICMI 2021), was co-publicity chair for FG2018, General chair of 3rd International Conference on Biometric Engineering and Applications (ICBEA 2019), and Late Breaking Results Co-Chairs of 21st ACM International Conference on Multimodal Interaction (ICMI 2019), has served as area chairs for several conferences and is associate editor for Pattern Recognition, IEEE Transactions on Circuits and Systems for Video Technology, and Image and Vision Computing Journals. She has lectured tutorials at ICPR 2006, ICCV 2009, SCIA 2013 and FG 2018, authored/edited three books and eight special issues in journals. Dr. Zhao was a Co-Chair of many International Workshops at ICCV, CVPR, ECCV, ACCV and BMVC. Her current research interests include image and video descriptors, facial-expression and micro-expression recognition, emotional gesture analysis, affective computing, and biometrics. Her research has been reported by Finnish TV programs, newspapers and MIT Technology Review.