IEEE Open Journal of the Computer Society

Received 26 March 2024; accepted 11 May 2024. Date of publication 29 May 2024;
date of current version 19 June 2024. The review of this article was arranged by Associate Editor Wei Liu.

Digital Object Identifier 10.1109/OJCS.2024.3406645

4D Feet: Registering Walking Foot Shapes Using Attention Enhanced Dynamic-Synchronized Graph Convolutional LSTM Network

FARZAM TAJDARI [1,2,3], TOON HUYSMANS[1,4], XINHE YAO[1], JUN XU [1], MARYAM ZEBARJADI [5] (Graduate Student Member, IEEE), AND YU SONG [1] (Member, IEEE)

[1]Faculty of Industrial Design Engineering, Delft University of Technology, 2628 CE Delft, The Netherlands
[2]Department of Mechanical Engineering, Dynamics and Control (D&C) Group, Technical University of Eindhoven, 5612 AZ Eindhoven, The Netherlands
[3]Department of Mechanical Engineering, Cognitive Robotic (CoR) Group, Delft University of Technology, 2628 CD Delft, The Netherlands
[4]Imec-Vision Lab, Department of Physics, University of Antwerp, 2610 Antwerp, Belgium
[5]Department of Electrical Engineering, University of Minnesota, Minnesota, MN 55455 USA

CORRESPONDING AUTHOR: FARZAM TAJDARI (e-mail: f.tajdari@tudelft.nl).

This work was supported by the Dutch NWO Next UPPS - Integrated design methodology for Ultra Personalised Products and Services Project under Grant 15470.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Human Research Ethical Committee of Delft University of Technology under Application No. 2445.

ABSTRACT 4D-scans of dynamic deformable human body parts help researchers have a better understanding of spatiotemporal features. However, reconstructing 4D-scans utilizing multiple asynchronous cameras encounters two main challenges: 1) finding dynamic correspondences among different frames captured by each camera at the timestamps of the camera in terms of dynamic feature recognition, and 2) reconstructing 3D-shapes from the combined point clouds captured by different cameras at asynchronous timestamps in terms of multi-view fusion. Here, we introduce a generic framework able to 1) find and align dynamic features in the 3D-scans captured by each camera using the nonrigid-iterative-closest-farthest-points algorithm; 2) synchronize scans captured by asynchronous cameras through a novel ADGC-LSTM-based-network capable of aligning 3D-scans captured by different cameras to the timeline of a specific camera; and 3) register a high-quality template to synchronized scans at each timestamp to form a high-quality 3D-mesh model using a non-rigid registration method. With a newly developed 4D-foot-scanner, we validate the framework and create the first open-access data-set, namely the 4D-feet. It includes 4D-shapes (15 fps) of the right and left feet of 58 participants (116 feet including 5147 3D-frames), covering significant phases of the gait cycle. The results demonstrate the effectiveness of the proposed framework, especially in synchronizing asynchronous 4D-scans.

INDEX TERMS 4D foot scanner, dynamic feature recognition, synchronized scans, LSTM network, non-rigid registration.

I. INTRODUCTION

Human movements often result in significant shape deformations of various body parts. The emergence and advancement of 4D scanning, capable of capturing 3D geometric shapes over time, have enhanced our understanding of this dynamic anthropometry [4], [8] and human body deformation while performing different types of activities. These insights can be valuable in a wide range of applications [36], [44]. Outcomes of research on 4D scans can be applied in many areas, e.g. building virtual avatars, performing (virtual) ergonomics evaluations, developing computer games, designing personal protective equipment, workwear, sportswear, and other practical garments [3].

To acquire 4D scans, multiple (depth) cameras are often used, a setup that presents several complications that need to be addressed for reliable results. These challenges include

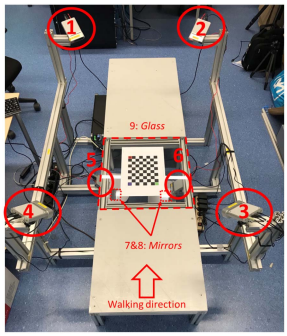


FIGURE 1. The TU Delft 4D foot Scanner with six Microsoft Azure Kinect DK cameras to capture the 4D foot shape.

temporal synchronization of captured frames to account for the asynchrony in data capture, accurate 3D space alignment, establishing feature correspondences across frames taken by different cameras and at different times, effectively fusing the data from multiple sources, and ensuring data consistency and accuracy throughout the reconstruction process. One solution to overcome this challenge may involve with hardware synchronisation approach; however, it is challenging to balance the needed resolutions of the images, the needed time duration, the buffer of the depth cameras, the data transfer rate, the computing power, and the storage [2]. For instance, it is difficult to balance the needed resolutions of the images, the needed time duration, the buffer of the depth cameras, the data transfer rate, the computing power, and the storage [2]. For instance, to capture 640 × 576 depth images by 6 cameras at 30 frames per second (fps), the needed bandwidth is about 2 Gb/s. This poses challenges in the design of a 4D scanning system, especially for a low-cost system. In addition, "dropped frames" are frequently observed in the captured data, mainly due to that the huge amount of to-be-transferred data leads to a nonlinear accumulative delay regarding each camera [30]. Relevant literature on 3D reconstruction from asynchronise multi-views studied in [20], [29], [31] show the importance of a general time-synchroniser method for collecting any 4D data-set through multiple cameras. In a practical case of using 6 Azure Kinect DK cameras for 15 fps 4D scanning, as in Fig. 1, even when all cameras are hardware synchronized, we found that there are on average 2 ms delays for each frame acquired by those cameras in a 3-seconds scanning session. Note that the delay is accumulative, i.e. at the beginning of the scanning all the cameras' outputs are well-aligned based on their clocks; however, the longer the duration of the scanning is, the more the delay accrues, resulting in a divergence of the geometry in each frame regarding the timestamps.

Therefore, a robust software synchronization algorithm is necessary to mitigate the mentioned complications. In the context of scanning human body (parts), a potential approach is to leverage the prior knowledge of human actions and the associated dynamic features to synchronize the captured frames. In the past decades, recognizing human dynamics features has attracted a lot of attention in the field of computer vision. The developed 3D human action recognition methods can be roughly classified as the RGB video-based approaches [54], [56], skeleton-based methods [39], [40], depth image-based methods [61], [63] and the point cloud-based method [58]. Although the existing methods are proven to be effective in many applications, e.g., video surveillance, human-computer interaction, sports analysis [36], [44], most of them are limited to employ (depth) images as the input, and the recognized 3D actions as the output. Extracting point-to-point correspondences among sequential point clouds from multiple views e.g., cameras, is rarely investigated [27], [38], [58] with core study on non-rigid tracking using depth cameras [25]. In this regard, a key constraint arises from the inherent limitation of a single-view range sensor, which hinders the acquisition of data in occluded regions, thereby yielding incomplete observations of 3D environments [25]. Consequently, prevalent non-rigid motion tracking techniques are confined to processing only the visible portions of a scene. Nevertheless, the imperative to deduce comprehensive motion patterns from partial observations is crucial for various high-level tasks. Thus, regarding 3D human action recognition, there are two fundamental challenges: 1) establishing the dynamic connectivity among asynchronous images (scans) captured by different cameras in terms of dynamic feature synchronization i.e. temporal correspondence, and 2) extracting meaningful dynamic features from the combined camera views for accurate analysis of deformation, i.e. multi-view fusion.

In this article, using a newly developed low-cost 4D foot scanner based on 6 Microsoft Azure Kinect DK depth cameras, we developed a framework to synchronize and register the captured asynchronous images on significant phases of the gait cycle, resulting in a new open-access 4D Feet data-set of 58 subjects (116 feet). Our main contributions are:

- Introducing a framework to synchronized spatiotemporal asynchronous scans captured from multiple cameras and to track a point's correspondences in all the frames to extract dynamic features of each vertex e.g, velocity;
- Establishing an adaptive correspondence point selection approach based on a nonrigid-Iterative-Closest-Farthest-Points (ICFP) algorithm between the 3D frames of one camera guarantees convergence to highest probability of finding useful points, known as "Synchronised Graph".
- Developing a novel Attention Enhanced Dynamic-Synchronised Graph Convolutional (ADGC)-LSTM network to synchronize the dynamic features extracted from different cameras besides existing algorithms;
- Presenting the first 4D mesh-morphed walking foot open-access data-set (4D Feet), as a validation of the proposed framework.

II. RELATED WORK

A. SKELETON-AND-DEPTH-BASED ACTION RECOGNITION

The skeleton-based approach and the depth-based approach are often used in recognizing dynamic features of human

actions based on prior knowledge [27]. Regarding skeleton-based 3D action recognition, sequence-based approaches, and graph-based approaches are often used. Via describing the skeleton as a sequence of joints, the sequence-based approaches [39], [40] employed the RNN (Recurrent Neural Network) based methods to extract temporal connectivity among those featured points. The graph-based approaches [23], [64] often utilized GCN (Graph Convolution Network) to exploit spatiotemporal connectivity by considering the skeletal structure as a graph, where the featured points are considered as the points of the graph. Regarding depth-based 3D action recognition, the available methods [33], [55] mainly use the visualization features through 2.5D depth maps. Although both approaches are able to give a reasonably good estimation of the 3D actions that the target subject performed, it is difficult to form a generic framework to fully extract dynamic features based on a few featured points of the moving object, which may cause a reduction in the performance of 3D action recognition.

B. 3D POINT CLOUDS ACTION RECOGNITION

Deep learning tools play a key role in extracting human actions via 3D point clouds, which is widely employed in recent studies [9], [12], [13], [16], [28], [37], [51], [57], [60]. Among them, C3D [51] utilizes a deep 3D CNN trained on extensive video datasets for spatial-temporal feature acquisition. I3D [9] utilizes 3D CNN for acquiring spatial-temporal features effectively. X3D [12] demonstrates comparable efficiency and efficacy in recognizing actions in videos. SlowFast [13] represents a 3D architecture integrating both slow and fast pathways to capture motion details. PointNet [37] employs a set of MLPs on each of the individual vertices to identify the unique features. Next, it utilizes a max-pooling layer to generate the global identifier for each point cloud which does not use any geometry-based connectivity of the local neighboring structure. Contrary to these single-frame-based point cloud analysis approaches, in this article, we present a simple and effective framework for time series 3D shape reconstruction and action recognition, in which we explicitly use temporal information in the motion stream to identify dynamic features.

C. NON-RIGID TRACKING USING DEPTH CAMERAS

Various techniques exist for tracking non-rigid objects, employing different adaptations of the Non-rigid Iterative Closest Point (N-ICP) algorithm [1], [45], [46], [47], [49], [66]. In this approach, the iterative minimization of point-to-point or point-to-plane distances for corresponding points is a common practice. To address issues like uncontrolled deformations and motion ambiguities, deformation regularizers such as As-Rigid-As Possible (ARAP) [43] or embedded deformation are often incorporated into the N-ICP optimization process. DynamicFusion [32] was among the early real-time methods that simultaneously tracked and reconstructed non-rigid surfaces. Building upon DynamicFusion, VolumeDeform [19] enhanced tracking robustness by introducing sparse SIFT feature matches. DeepDeform [7] leverages deep learning to replace classical feature matching with CNN-based correspondence matching. Li et al. [24] took a step further by differentiating through the N-ICP algorithm, yielding a dense feature matching term. Neural Non-Rigid Tracking [6] shares a similar approach but emphasizes end-to-end robust correspondence estimation. For handling topology changes, KillingFusion [41] directly estimates the motion field based on a pair of Signed Distance Fields (SDF). While existing methodologies predominantly focus on the visible closest features shared in all scenes, our approach takes a stride beyond by converging to highest probability of finding useful features that are not necessarily the closets ones.

III. MATERIALS AND DATA COLLECTION

A 4D foot scanner was developed at TU Delft [22] for acquiring dynamic foot shape data. Fig. 1 presents the next generation of the 4D foot scanner which utilizes six Microsoft Azure Kinect DK cameras to capture the 4D foot shapes, where four cameras are installed on the top (id 1, 2, 3, and 4) and two cameras are at the bottom (id 5, and 6). To adapt to the minimal focal distance ($\sim$ 50 cm) of the cameras at the bottom, two first-surface mirrors (id 7 and 8) were placed on the floor to "fold" the optical path for lowering the height of the scanner for a better user experience. A 9 mm thickness plexglass (id 9) was installed on the footpath to enable capturing the shape of the bottom of the foot while a subject is walking.

The spatial positions and orientations of all cameras were optimized to maximize the resolutions of the captured scans and the intersections of effective view volumes of 6 cameras [22]. To transform the captured data to a global coordinate system, we used a two-sided checkerboard shown in Fig. 1, and the code in [15] is utilized.

IV. METHODOLOGY

In this section, we present the overall workflow of our proposed framework for synchronizing captured frames from different cameras, as depicted in Fig. 2. The first step involves establishing correspondences for dynamic features. After scanning, each camera generates a set of time series 3D point clouds, and initially, there are no logical correspondences among them. To address this issue, we have developed a novel extended version of the Nonrigid Iterative Closest-Farthest Points (ICFP) scheme to establish these correspondences. Subsequently, after aligning the data from all cameras, we have designed a new network called ADGC-LSTM to synchronize the aligned data from different cameras with each other. Finally, we introduce the mesh registration method to along synchronized scans of different cameras at each timestamp. We provide detailed information about these algorithms in the subsequent sections.

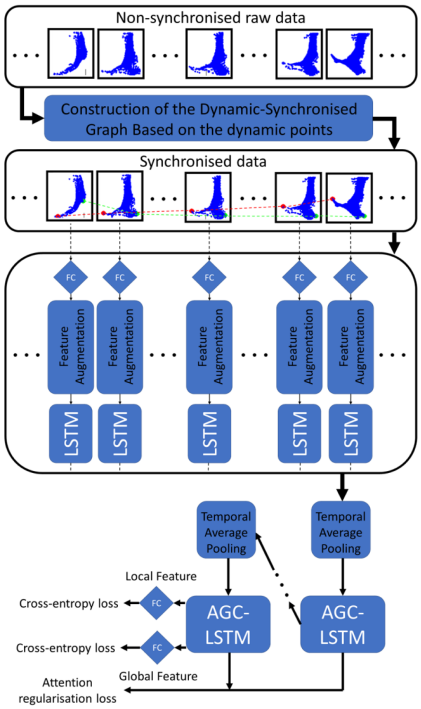


FIGURE 2. **The overall proposed workflow for time synchronization. The framework includes establishing correspondence for dynamic features, synchronizing correspondences from different cameras, and mesh registration of them.**

A. CONSTRUCTION OF THE DYNAMIC-SYNCHRONISED GRAPH BASED ON THE DYNAMIC POINTS

After scanning, each camera gives a set of time series 3D point clouds, and there are no logical correspondences among them. This prevents us from explain any dynamic features between the frames as the correspondence of points from one frame to the other frame, known as dynamic points, does not exist. Fig. 3 (the first row of each sub-figure) presents this "lack of correspondence", where we selected two points (highlighted with red and green colors) in the first frame of each camera, and tracked these points in the rest frames using point IDs in the acquired point clouds. To be able to have meaningful dynamic features between frames of a camera needed as the key nodes used for the ADGC-LSTM network in Section 2, we established the correspondences of points using a novel extended version of the Nonrigid Iterative Closest-Farthest Points (ICFP) scheme [48] which guarantees to find proper corresponded points in a limited number of iterations from a Source mesh ($\mathbb{S}$) to a Target mesh ($\mathbb{T}$).

In the process of finding correspondences from each point on $\mathbb{T}$ to $\mathbb{S}$, initially, each point on $\mathbb{S}$ may have multiple corresponding points on the $\mathbb{T}$. In this case, we logically select either the closest or the farthest point. In each iteration of the registration process, a boundary distance (l) in (1) is defined as the corresponding distance matrix from $\mathbb{T}$ to $\mathbb{S}$.

$$l = m + \zeta\sigma \tag{1}$$

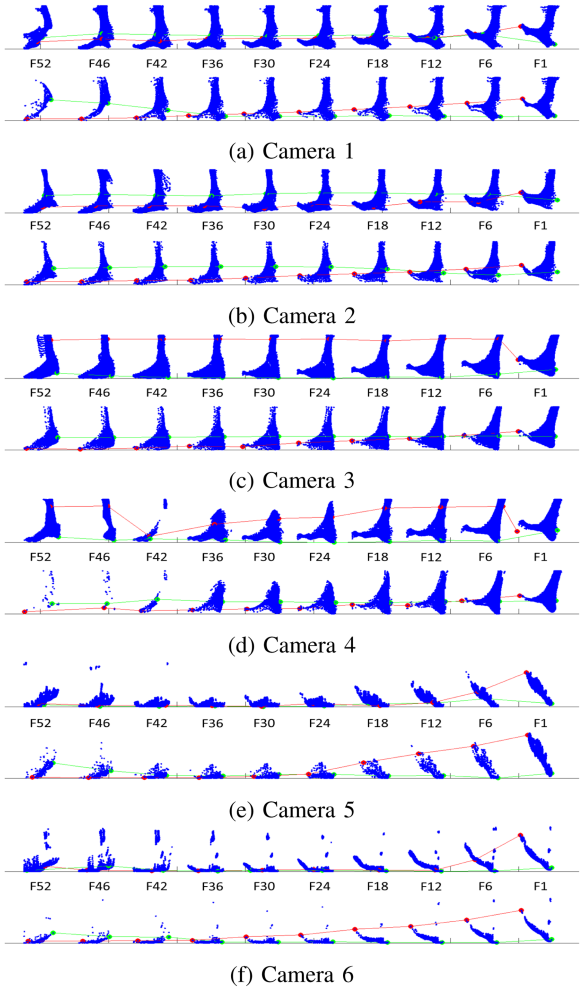


FIGURE 3. **(a) to (f) show data from 6 cameras and for each camera, Top: Raw data; Bottom: Dynamic-Synchronised Graph as the key nodes.**

where m and σ are the mean and standard deviation of the counted distances for the correspondences from $\mathbb{T}$ to $\mathbb{S}$, respectively. ζ is the probability indicator in [48] regulates the closest-farthest point selection and has an acceptable range in [0, 3.09] refer to Table 3 of the book in [14]. In [48], the ζ is a predefined constant variable that is numerically defined based on registering a source foot on only one target foot to minimise a concave parabolic cost function (J) including two terms of percentage of mean mesh quality loss and percentage of the target vertices involved in the nonrigid registration. However, there is no guarantee that the selected ζ results in the minimization of the cost function for any other target foot. Thus, here we extend the corresponding selection criterion by designing an adaptive ζ finds the minimum cost function by iterations and can be implemented on any other registering shapes. Assuming J from [48]

$$\min J = \sum \frac{|\bar{Q}^{final} - \bar{Q}^{0}|}{\bar{Q}^{0}} + \frac{N_{in}^{T}}{N_{tot}^{T}} \tag{2}$$

where $\bar{Q}^{final}$ and $\bar{Q}^{0}$ are the average of mesh quality for all vertices on the source mesh before and after registration

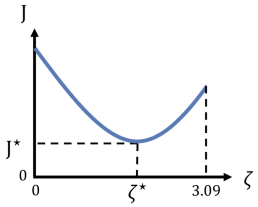


FIGURE 4. Assumed $J-\zeta$ shape.

respectively. Also, N_{in}^{T} is the number of vertices from target employed as corresponding points during the nonrigid registration process, and N_{tot}^{T} is the total number of vertices on the target mesh.

1) ASSUMPTION

For the design of the estimator, we formulate a parabolic $J-\zeta$ relationship. In particular, we employ the following function describing the $J-\zeta$ relationship, also depicted in Fig. 4,

$$J = a\zeta^2 + b\zeta + c, \tag{3}$$

where $a \in \mathbb{R}_{>0}$ and $b \in \mathbb{R}_{>0}$ are unknown parameters, and C is equal to $J(0)$ defined as initial condition of J which is assumed known; function (3) has a minimum point $(\zeta^\star, J^\star)$ as

$$\zeta^\star = \frac{-b}{2a}; \qquad J^\star = \frac{-b^2}{4a} + J(0). \tag{4}$$

2) ADAPTIVE ζ DESIGN

By replacing the nominal values of $J^\star$ and $\zeta^\star$ in (3), the error of J from $J^\star$ is

$$J - J^\star = a\left(\zeta^2 - \zeta^{\star 2}\right) + b\left(\zeta - \zeta^\star\right). \tag{5}$$

Now, we introduce the integral error states

$$E_J = \int \left(J - J^\star\right) dt; \quad E_\zeta = \int \left(\zeta - \zeta^\star\right) dt, \tag{6}$$

allowing to define the integral error system

$$\dot{E}_J = J - J^\star; \quad \dot{E}_\zeta = \zeta - \zeta^\star, \tag{7}$$

that can be reformulated as

$$\dot{x} = B_e u_e + r_e, \tag{8}$$

where

$$x = \begin{bmatrix} \int J\, dt \\ \int \zeta\, dt \end{bmatrix}, u_e = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \zeta^2 - \zeta^{\star 2} \\ \zeta - \zeta^\star \end{bmatrix} \tag{9}$$

$$B_e = \begin{bmatrix} a & b \\ 0 & 1 \end{bmatrix}, r_e = \begin{bmatrix} J^\star \\ \zeta^\star \end{bmatrix}. \tag{10}$$

We propose controlling system (8) using MRAC [42], which let us simultaneously identify the unknown parameters a and b (both appearing in B_e) and minimise the tracking error. In order to proceed, we introduce the feedback control law (see, e.g., Chapter 1 in [53])

$$u_e = -\hat{\Pi}(x - r_e), \tag{11}$$

where $\hat{\Pi}$ is an unknown matrix that needs to be estimated. We then introduce a model reference

$$\dot{x}_M = -A_M x_M + B_M r_e, \tag{12}$$

where A_M is designed as a positive definite matrix and B_M is an arbitrarily defined matrix in which they guarantee stable model reference dynamics. Let us define the error between the integral states and the model reference $e = x - x_M$, whose dynamics are defined as (see [42])

$$\begin{aligned} \dot{e} &= \dot{x} - \dot{x}_M \\ &= B_e\left(-\hat{\Pi}x + \hat{\Pi}r_e\right) + r_e + A_M x_M - B_M r_e + A_M x - A_M x \\ &= -A_M(x - x_M) + B_e\left(-\hat{\Pi} + A_M\right)x + B_e\left(\hat{\Pi} - \frac{B_M - I}{B_e}\right) r_e \\ &= -A_M e + B_e\left(-\hat{\Pi} + A_M\right)x + B_e\left(\hat{\Pi} - \frac{B_M - I}{B_e}\right) r_e \end{aligned} \tag{13}$$

Knowing $v = \begin{bmatrix} x \\ r_e \end{bmatrix} \in \mathbb{R}^{4\times 1}$, the error dynamic system is

$$\begin{aligned} e &= -A_M e + \begin{bmatrix} -B_e\hat{\Pi} + B_e A_M & B_e\hat{\Pi} - B_M + I \end{bmatrix} v \\ &= -A_M e + \tilde{\phi} v \end{aligned} \tag{14}$$

where $\tilde{\phi} = [-B_e\hat{\Pi} + B_e A_M \quad B_e\hat{\Pi} - B_M + I] \in \mathbb{R}^{2\times 4}$. We assume that $\tilde{\phi} = \hat{\phi} - \phi$, where $\hat{\phi} \in \mathbb{R}^{2\times 4}$ is an estimating matrix, namely $\hat{\phi} = [-B_e\hat{\Pi} \quad B_e\hat{\Pi}]$ and $\phi \in \mathbb{R}^{2\times 4}$ is the unknown constant matrix, namely $\phi = [-B_e A_M \quad B_M - I]$ defines $\dot{\phi} = 0_{2\times 4}$. Thus,

$$\dot{\tilde{\phi}} = \dot{\hat{\phi}}. \tag{15}$$

We can observe that the error dynamic of (14) is bounded over time if $\tilde{\phi}$ is bounded, and the error is asymptotically stable if $\tilde{\phi}$ converges to zero, considering that $-A_M$ is selected as a stable matrix with negative eigenvalues, while B_M and B_e are constant matrices.

In order to study the convergence of $\tilde{\phi}$ to zero, a Lyapunov function $\mathcal{V} \in \mathbb{R}^{2\times 2}$ is employed as follows:

$$\mathcal{V} = e\mathcal{P}e^\top + \tilde{\phi}\Gamma^{-1}\tilde{\phi}^\top, \tag{16}$$

where $\mathcal{P} \in \mathbb{R}_{>0}$ and $\Gamma \in \mathbb{R}_{>0}^{4\times 4}$ imply that $\mathcal{V} > 0$. In order to guarantee stability, it is sufficient if $\dot{\mathcal{V}} \leq 0$, then

$$\frac{d\mathcal{V}}{dt} = \dot{e}\mathcal{P}e^\top + e\mathcal{P}\dot{e}^\top + \dot{\tilde{\phi}}\Gamma^{-1}\tilde{\phi}^\top + \tilde{\phi}\Gamma^{-1}\dot{\tilde{\phi}}^\top. \tag{17}$$

By replacing $\dot{e}$ from (14), and considering (15), we obtain

$$\frac{d\mathcal{V}}{dt} = -A_M e\mathcal{P}e^\top - e\mathcal{P}e^\top A_M^\top + 2e\mathcal{P}v^\top\tilde{\phi}^\top + 2\dot{\hat{\phi}}\Gamma^{-1}\tilde{\phi}^\top. \tag{18}$$

As A_M is positive definite, $-A_M e\mathcal{P}e^\top - e\mathcal{P}e^\top A_M^\top$ is negative semi-definite matrix, thus $\frac{d\mathcal{V}}{dt} \leq 0$ if and only if

$$2e\mathcal{P}v^\top\tilde{\phi} + 2\dot{\hat{\phi}}\Gamma^{-1}\tilde{\phi}^\top = 0, \tag{19}$$

which is a sufficient condition for stability where the changes in the estimating unknown matrix $\hat{\phi}$ is

$$\dot{\hat{\phi}} = -e\mathcal{P}v^{\top}\Gamma, \tag{20}$$

where Γ is known as the *growth rate* of the estimation law. Using (20), the instruction of ϕ and $\hat{\phi}$, and $\tilde{\phi} \to 0$, we may conclude that $\hat{\Pi} \to A_M$, and $B_e \to (B_M - I)\hat{\Pi}^{-1}$, which results in $\hat{\zeta} \to \zeta^{\star} = \frac{-B_{e_{1,2}}}{2B_{e_{1,1}}}$ from (4), and (10). Thus, having $\hat{\zeta} \to \zeta^{\star}$ and for a point on $\mathbb{S}$, if a number of points on $\mathbb{T}$ are selected, we consider the point with the largest distance among the selection, if all the distance for the population is greater than the l. Otherwise, we select the closest point as the corresponding point to the point on $\mathbb{S}$. The ICFP scheme was used to find the available correspondences for all frames captured by a single camera e.g. top-rows of the sub-figures in Fig. 3. In the implementation, we use the ICFP to match each consequent pair of frames (e.g. i^{th} frame and $(i+1)^{th}$ frame), starting from the first frame to the last frame, e.g. for 100 captured frames, 99 pairs were used to generate the correspondences matrix. Apparently, not all points in a frame have correspondences in the neighboring frames, as a new frame may not be able to capture all the points captured in the previous frame e.g. from comparing Fig. 3(d)-top with Fig. 3(d)-bottom after the 42^{th} frame (F42) the density of Dynamic-Synchronised Graph is reduced. Thus, some frames with very low density point-clouds are skipped due to the lack of correspondences.

B. TIME SYNCHRONIZATION

After establishing the correspondences of dynamic features in previous section, an end-to-end network based on ADGC-LSTM for points-network-based action behavior recognition, is explained provided to time synchronization. At first, we review the general architecture of the network and then in the next subsection provide our suggested ADGC-LSTM network.

1) ADGC-LSTM NETWORK

In the analysis of sequential geometric shapes, many studies suggested that the LSTM, as a transformation of RNN, has a strong capability to understand long-term time dependency of the phenomenons e.g., understanding temporal dynamics of limited points-network (skeleton) sequences. However, using LSTM alone is difficult for incorporating spatial relations in the limited points-network-based action recognition. To this end, AGC-LSTM [40], as an extension of LSTM, was developed to incorporate not only unique features of spatial configuration and temporal dynamics but also the coincident relationships between the spatial domain and temporal domain.

In the process of capturing moving objects (4D scanning), the requirements of the needed movement ranges and the limited views of the cameras are always contradictory factors. This often results in a compromise in the design of the 4D scanner, either with a very small working envelope

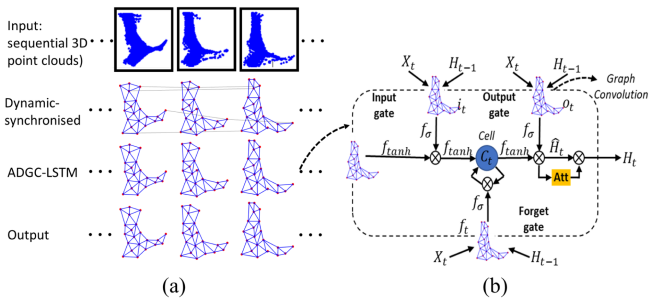


FIGURE 5. The structure. (a) One ADGC-LSTM layer; (b) One ADGC-LSTM unit adapted from [40].

with limited movements or with sparse points in some of the captured point clouds. Commercial systems may employ more camera modules in 4D scanning; however, at the cost of investment and increased complexity. As the human movements are part of nature and cannot be constrained in a limited range, we target at building correspondences between/among sparse point clouds. Therefore the principles of the graph convolution model which has been broadly employed in sequential data with limited points-network nodes were adopted. Establishing the graph model plays a fundamental role in the graph convolution algorithm. Available graph convolution models e.g, AGC-LSTM, have several limitations for example using single graph structures, illed-correspondences among points, and inadequate discrimination of dissimilar actions. Here we develop a graphic model according to the Dynamic-Synchronised Graph based on the dynamic points, aiming at generating more sparse dynamic features to enhance the capability of the AGC-LSTM model in classifying spatiotemporal features and improve the precision of the action recognitions. The proposed novel method presented here is named Attention Enhanced Dynamic-Synchronized Graph Convolutional (ADGC)-LSTM network. The details of the method are presented below.

Following the structure of LSTM, the ADGC-LSTM includes three gates: the input gate i_t, forgetting gate f_t, and output gate o_t. The input X_t, hidden state H_t, and cell memory C_t are graph structure data, and the graph structure is generated by the ICFP (Nonrigid Iterative Closest-Farthest Points) algorithm explained in Section IV-A. The graph convolution operator in the ADGC-LSTM, cell memory C_t, and hidden state H_t can be used to extract temporal dynamics, and include spatial structure information. Fig. 5(a) describes the structure of an ADGC-LSTM layer. Fig. 5(b) describes the structure of the ADGC-LSTM unit. Equation (21) describes the functions of the ADGC-LSTM unit.

$$\begin{aligned}
i_t &= \sigma\left(W_{xi} \star gX_t + W_{hi} \star gH_{t-1} + b_i\right)\\
f_t &= \sigma\left(W_{xf} \star gX_t + W_{hf} \star gH_{t-1} + b_f\right)\\
o_t &= \sigma\left(W_{xo} \star gX_t + W_{ho} \star gH_{t-1} + b_o\right)\\
u_t &= \tanh\left(W_{xc} \star gX_t + W_{hc} \star gH_{t-1} + b_c\right)\\
C_t &= f_t \odot C_{t-1} + i_t \odot u_t
\end{aligned}$$

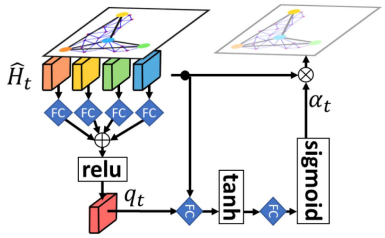


FIGURE 6. Illustration of spatial attention mechanism principal adapted from [40].

$$\hat{H}_t = o_t \odot \tanh(C_t)$$
$$H_t = f_{att}\left(\hat{H}_t\right) + \hat{H}_t \tag{21}$$

where $\star g$ defines the graph convolution operator and $\odot$ defines the Hadamard product. $\sigma(.)$ denotes the sigmoid activation function. u_t denotes the modulated input. $\hat{H}_t$ explains an intermediate hidden state. $W_{xi} \star gX_t$ defines a graph convolution of X_t with W_{xi}. The used graph convolution is the same as the graph convolution employed for the Graph Convolutional Neural (GCN) network in [62] with K number of labels. $f_{att}(.)$ is an attention network that can select the diverse information of key nodes. The output $H_t$reinforces the information of key nodes, without neglecting the information of non-focus nodes, aiming at better integrity of spatial information.

The ADGC-LSTM network logically insists on key nodes by using a soft attention mechanism that automatically quantifies the emphasis level of the key nodes. The importance of the spatial attention network is depicted in Fig. 6. The intermediate hidden state ($\hat{H}_t$) of ADGC-LSTM contains persistent spatial structure information and temporal dynamics. The state practically improves the selection of the key nodes procedure. In order to guarantee that independent degree weights are established and reinforce the significance of dissimilar nodes for dissimilar types of actions, we employed a query feature as:

$$q_t = \text{relu}\left(\sum_{i=1}^{N} W\hat{H}_{t_i}\right) \tag{22}$$

where W defines the trainable parameter matrix, and N is the number of nodes in the graph. Thus the attention scores of all nodes would be specified as:

$$\alpha_t = \text{sigmoid}\left(U_s \tanh\left(W_h\hat{H}_t + W_q q_t + b_s\right) + b_u\right) \tag{23}$$

where $\alpha_t = (\alpha_{t_1}, \alpha_{t_2}, \ldots, \alpha_{t_N})$, and U_s, W_h, and W_q are the trainable matrices. b_s and b_u are the bias. A non-linear function *sigmoid* is employed regarding the probability of selected key joints. The hidden state H_{t_i} of node v_{t_i} is considered as $(1 + \alpha_{t_i})\hat{H}_{t_i}$. The attention enhanced hidden state H_t is considered as an input for the next ADGC-LSTM layer. In the final layer of the ADGC-LSTM network, the accumulation of all node features is classified as a global feature F_t^g, and the weighted sum of focused nodes is classified as a local feature F_t^l:

$$F_t^g = \sum_{i=1}^{N} H_{t_i}; \quad F_t^l = \sum_{i=1}^{N} \alpha_{t_i}\hat{H}_{t_i}. \tag{24}$$

For Graph Model Based on Human feet dynamic points, firstly a linear layer and LSTM layer were employed to convert the 3D coordinate of each key node into a high-dimensional feature space regarding the key node-network sequence. The preliminary linear layer maps the 3D coordinates onto a 256-dimensional vector, as the geometric features P_t, i.e., P_{ti} defines the geometry feature of key node i. As it includes only geometry information, P_{ti} is effective to proceed with the learning process regarding spatial structure features in graph models. The differential feature V_{ti} between two sequential frames, facilitates the dynamic feature understanding used to train the ADGC-LSTM. The sequential group of features is able to explain a more sparse domain of feature information better, while the differential of the features is more sensitive to the changes of the feature vectors. Thus, the LSTM layer was utilized to avoid having unnecessary sensitivity between the sequential feature groups. Equation (25) presents this proposition.

$$\begin{aligned} E_{ti} &= f_{lstm}\left(concat\left(P_{ti}, V_{ti}\right)\right) \\ &= f_{lstm}\left(concat\left(P_{ti}, P_{(t-1)i}\right)\right) \end{aligned} \tag{25}$$

where E_{ti} is the augmented featured of key node i at time t.

Finally, the global feature F_t^g and local features F_t^l at each timestamp were converted to scores o_t^g and o_t^l of each class. According to (21), the predicted probability of the i^{th} class can be obtained as:

$$\hat{y}_{ti} = \frac{e^{o_{ti}}}{\sum_{j=1}^{C} e^{o_{ti}}}, i = 1, \ldots, C \tag{26}$$

In the training process, taking into account the hidden state of each time interval, the ADGC-LSTM includes short-term dynamics and the loss function with the structure in (27), extracted to the train model as:

$$\begin{aligned} L = &-\sum_{t=1}^{T_3}\sum_{i=1}^{C} y_i \log\hat{y}_{ti}^g - \sum_{t=1}^{T_3}\sum_{i=1}^{C} y_i \log\hat{y}_{ti}^l \\ &+ \bar{\lambda}\sum_{j=1}^{3}\sum_{n=1}^{N}\left(1 - \frac{\sum_{t=1}^{T_j}\alpha_{tnj}}{T_j}\right)^2 \\ &+ \bar{\beta}\sum_{j=1}^{3}\frac{1}{T_j}\sum_{t=1}^{T_j}\left(\sum_{n=1}^{N}\alpha_{tnj}\right)^2 \end{aligned} \tag{27}$$

where $y = (y_1, \ldots, y_c)$ is the ground-truth label. T_j denotes the number of time intervals on the j^{th} ADGC-LSTM layer. The third term is considered to emphasize equally to variation of featured points. The final term is to restrict the number of interested nodes. $\bar{\lambda}$ and $\bar{\beta}$ are weight decaying coefficients. We performed thorough parameter adjustment during the design process of our ADGC-LSTM network in order to optimize its

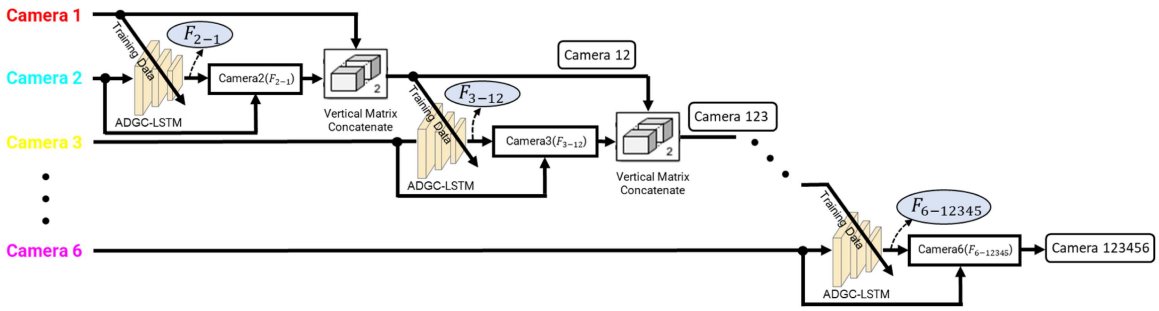


FIGURE 7. Hierarchical learning-synchronization process.

performance. The number of layers, units per layer, learning rate, and dropout rate were among the hyperparameter combinations that we experimented with Section V-C. By applying this iterative process, we identified that a network architecture comprising three ADGC-LSTM layers with 512 units each, a learning rate of 0.001, and dropout regularization consistently outperformed other configurations on our validation set. This choice was further supported by our review of previous studies, which indicated that similar architectures had achieved promising results in related tasks e.g., [40].

2) ADGC-LSTM NETWORK FOR 4D FOOT SCANNING

To synchronize the captured frames between each pair of cameras, we use the generated Dynamic-Synchronised Graph of each camera. We use one camera's Dynamic-Synchronised Graph as the supervisor to train our ADGC-LSTM Network and the other one for validation. In this case, we use a hierarchical learning process to have the maximum overlap between cameras with the shown framework in Fig. 7. In the figure, firstly we synchronize Camera 2 with Camera 1 (where the corresponding frames of Camera 2 to Camera 1 is F_{2-1}) and name the overall point cloud as Camera 12 (Camera 1 with synchronized Camera 2). Then synchronize Camera 3 with Camera 12 (with frame set of F_{3-12}) and name it as Camera 123. Then we continue with Camera 4, Camera 5, and finally Camera 6, to have all the cameras synchronized based on Camera 1.

C. MESH REGISTRATION

Based on the established correspondences, a cost function based on Tajdari et al. [47] is defined for registering meshed at each time step. Tajdari et al. [47] proposed the non-rigid registration formulation as a combination of distance (W, D, U), stiffness (M, G), and semi-curvature (W_c, A_c, B_c) terms summarised in the following formula

$$E(X) = \left\| \begin{bmatrix} \alpha M \otimes G \\ WD \\ \beta W_c A_c \end{bmatrix} X - \begin{bmatrix} 0 \\ WU \\ \beta W_c B_c \end{bmatrix} \right\|_F^2 = \|AX - B\|_F^2 \quad (28)$$

where, The sparse matrix D is formed to facilitate the transformation of the source vertices with the individual transformations contained in X via matrix multiplication, and denoted as $D = diag(v_1^T, v_2^T, \ldots, v_n^T)$, where $v_i \in \mathbb{S}$ and $i = 1, \ldots, n$, and n is the number of vertices on the $\mathbb{S}$. W is a diagonal matrix consisting of weights w_i. α is the stiffness constraint. To regularise the deformation, an additional stiffness term is introduced. Using the Frobenius norm $\|.\|_F$, the stiffness term penalizes the difference of the transformations of neighboring vertices, through a weighting matrix $G = diag(1, 1, 1, \gamma)$. During the deformation, γ is a parameter to stress differences in the skew and rotational part against the translation part of the deformation. The value of γ can be specified based on data units and the types of deformation [1]. The node-arc incidence matrix M (e.g. Dekker [11]) of the template mesh topology is employed to convert the stiffness term into the matrix form. As the matrix is fixed for directed graphs, the construction is one row for each edge of the mesh and one column per vertex. To establish the node-arc incidence matrix of the source topology, the indices (i.e. the subscripts) of edges and vertices are addressed, for any edge of r which is connected to vertices (i, j), in r^{th} row of M, and the nonzero entries are $M_{ri} = -1$ and $M_{rj} = 1$.

V. EXPERIMENT SETUP

A. DATA-SET

1) OUR DATA-SET

Using the proposed 4D scanner and the novel framework, we tried to build an open-access data-set of 4D feet data regarding significant phases of the gait cycle such as initial contact, foot flat, midstance, heel lift, and toe-off. An experiment was designed and approved by the local human research ethical committee. In the experiment, after a brief explanation, participants first read and signed the consent forms. Subjects under 18 had their consent forms signed by their parents/legal guardians. Then each subject was guided to walk through the glass bridge with his/her bare feet twice regarding the left and the right feet, respectively. Both feet of 59 subjects (26 females (♀) and 33 males (♂)) were scanned while the data of participant 53 was not saved and was excluded from the data-set, resulting in a data-set with 58 subjects. Among them, 55 subjects are right-handed and the rest are left-handed. The age of the population ranges from 6 to 50 years old where

TABLE 1. The Anthropometric Data

Sex	Age	Shoe size	Height	Weight	BMI
♀	24.0±5.1	37.5±1.5	161.1±9.0	55.9±9.4	21.5±2.6
♂	26.2±6.4	42.9±1.6	178.8±8.8	73.3±11.7	22.9±3.0

the mean age is 24 for females (♀) and 26.2 for males (♂). Their normal shoe sizes range from 32 to 46 (European sizes, 20-29.3 CM). To be more inclusive and address the diversity of the population, we invited subjects from different countries such as The Netherlands, Belgium, Italy, Spain, Latvia, Slovenia, Swaziland, Turkey, Iran, India, Thai, China, Japan, Costa Rica, Mexico, Cameroon, Nigeria. The anthropometric data of the population can be found in Table 1.

2) DATA-SET FOR REGISTRATION

In the experiment, both the right and the left feet shapes in data-set number 25 in the SHREC'14 data-set [35] were selected as the source surface. Before the experiment, the meshes of both feet were pre-processed for a more uniform mesh using ACVD, a freely available software provided by Valette et al. [52]. The acquired two meshes, each has 5000 vertices, were used as the inputs of the experiment as the source meshes for the nonrigid registration regarding the left and the right foot, respectively.

B. METHODS FOR COMPARISON

We compare the proposed methods in the framework with the following methods with similar state-of-the-art [17]:

- *ARIMA [59]:* Auto-Regressive Integrated Moving Average method is one of the well-known methods to anticipate the future values in a time sequential data-set.
- *VAR [65]:* Vector Auto-Regressive finds the pairwise connectivity between time-sequential data-sets.
- *LSTM [18]:* Long-Short Term Memory network, is a variant of RNN network.
- *GRU [10]:* Gated Recurrent Unit network, is a specific RNN network.
- *STGCN [62]:* A Spatial-Temporal Graph Convolution model is developed based on automatic learning of both the spatial and temporal patterns.
- *GeoMAN [26]:* A multi-level attention-based RNN model aimed for the geo-sensory time sequential anticipation problem.

Root mean square error (RMSE) of the geometry based on closest points is used as the metric.

C. ADGC-LSTM PARAMETERS' CONFIGURATION

In the experiments, a fixed length of $T = 40$ is used in (27) from each graph sequence as the input. Regarding the ADGC-LSTM, we assumed the neighbor set of each node includes only nodes directly connected with itself. Regarding a fair comparison with ST-GCN [62], the graph labeling function in ADGC-LSTM divides the neighbor set into $K = 3$ subsets

TABLE 2. RMSE Results of the Comparison Based on Closest Points Geometry Distance (CPGD), and Percent of Improvement (PI) Comparing to the Raw Data, for the Left and Right Foot

Method	Left foot		Right foot	
	CPGD (cm)	PI(%)	CPGD (cm)	PI(%)
raw data	7.01	–	7.35	–
ARIMA	6.81	2.8	5.95	19.1
VAR	3.24	53.8	4.22	42.6
LSTM	1.62	76.9	1.58	78.5
GRU	1.71	75.6	1.59	78.3
STGCN	1.21	82.7	1.14	84.4
GeoMAN	1.13	83.8	1.03	85.9
Our	**0.64**	**90.8**	**0.72**	**90.2**

according to [62]. In the training process, the Adam optimizer [21] is employed to optimize the network. Dropout with a probability of 0.5 is employed to prevent over-fitting on each participant's dataset. The parameters of $\bar{\lambda}$ and $\bar{\beta}$ are set to 0.01 and 0.001, respectively. We set the initial learning rate to 0.0005 which is reduced in every 15 epochs by multiplying 0.1 to the learning rate. In addition, we discretize the parameter estimation formula in (20) by considering $\dot{\hat{\phi}} = \frac{\hat{\phi}(k+1)-\hat{\phi}(k)}{\Delta k}$, $k \in \mathbb{Z}_{\geq 0}$; then, knowing that $\Delta k = 1$ as k is a sequentially increasing index (the index of intervals in the registration process), the estimation rule (20) turns into

$$\hat{\phi}(k+1) = \hat{\phi}(k) - e(k)\mathcal{P}v^{\top}(k)\Gamma, \tag{29}$$

where we assumed $\Gamma = 0.8I_{4\times 4}$, and $\mathcal{P} = 1$. Regarding the used mesh registration method in Section IV-C, we use the same parameter values in [47] regarding (28).

VI. RESULTS

A. MOTION SYNCHRONISATION

To evaluate the effectiveness of the synchronization, we developed a K-fold-like scheme where: 1) we used the mean Closest Points Geometry Distance (CPGD) [1] values between adjacent point clouds as the metric and 2) for each camera, we compared its synchronized scans to the merged results of other 5 cameras at each timestamp. That is, in the i^{th} frame and after synchronization with each of the aforementioned methods in Section V-B, we exclude the j^{th} camera points from the complete foot and calculate the CPGD of the camera j^{th} points with the remaining points. We repeat this process for all other cameras and the average values of errors are presented in Table 2. According to the table, our proposed method outperforms all the other methods for the both the left and right feet in the data-sets.

According to Table 2 and Fig. 8, one can be seen is that generally the output of the non-deep learning methods e.g., ARIMA and VAR, demonstrate a higher error than the deep learning methods e.g., LSTM, GRU, STGCN, GeoMAN. This is investigated numerically and the results are presented in Table 2, which shows that the deep-learning methods could

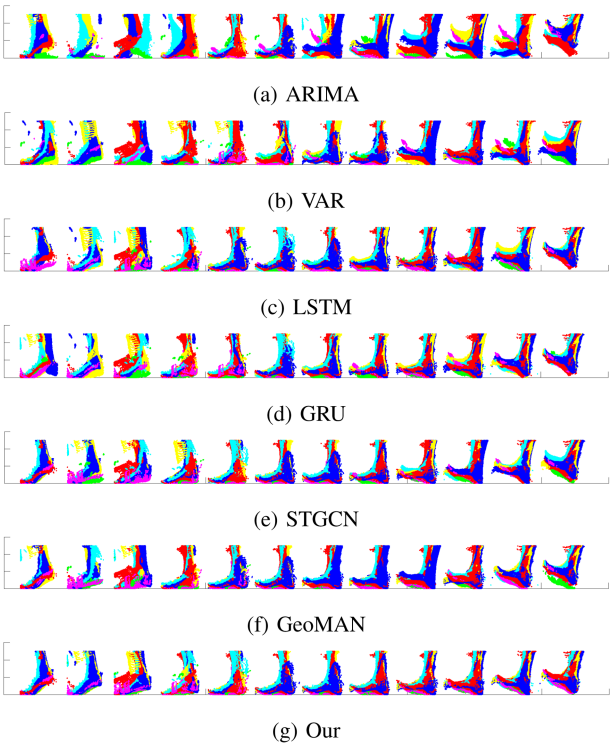


FIGURE 8. Results of time synchronisation with different methods.

averagely improve the performance for about 80% in terms of PI, revealing the limited abilities of the non-deep-learning methods to tackle non-linearity and complexity in time series analysis. Among the deep-learning methods, the models that simultaneously consider temporal and spatial correlations, e.g. STGCN, GeoMAN, and the proposed method, outperform other deep-learning-based methods including LSTM and GRU for about 11% in terms of PI. Where, GeoMAN slightly outperforms STGCN in terms of PI, defining that the multi-level attention mechanisms employed in GeoMAN enhance finding the correlation among dynamic features of the feet. Our ADGC-LSTM network, achieved better results than other included state-of-the-art methods, confirming the performance of the proposed method in describing spatial-temporal features of the walking foot.

B. ESTABLISHING REGISTRATION-BASED DATA-SET

Through the explained method in Section IV-C, we register the synchronised frames in Section V-A2 at each time step to establish a mesh morphed 3D geometry as Fig. 9.

In this regard, we can track not only the geometry of any point but also the dynamic features of the point such as velocity, and acceleration. In addition, we can compare the geometry or dynamic features of any points among all captured feet shapes. To this end, we numerically investigated the deformation variation of a few well-known foot dimensions in Table 3. The dimensions are length (L_f), width (W_f), and ball width (BW_f) according to [50] used in [48], and their variations are ΔL_f, ΔW_f, and ΔBW_f. Where the operator Δ defines the differences between the maximum and minimum

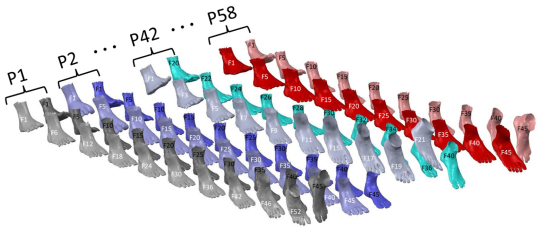


FIGURE 9. 4D Feet. We present a new 4D data-set of 58 Participants (P1, ···, P58 in the figure), including 5147 frames of 3D scans. The raw 3D scans (meshes) were collected at 15 fps through a novel 4D foot scanner including 6 Azure Kinect DK cameras. Then we showed how to synchronize the cameras through a novel deep-learning-based framework, and establish a mesh-morphed data-set.

TABLE 3. Foot Dimensions Variation Results

Parameter	Left foot	Right foot
ΔL_f (cm)	0.71±0.60	0.69±0.62
ΔW_f (cm)	1.1±0.9	1.0±0.9
ΔBW_f (cm)	1.2±0.9	1.1±1.0

value of the dimension for a participant during walking. By calculating the average foot length (L_{ave}) of all the feet in our data-set (both left and right feet) as 24.3 cm, we can see from Table 3 that the variation of L_f is about 3% of L_{ave}, and W_f and BW_f are about 5% of L_{ave}, which are a considerable variation and highlights the importance of 4D scanning, and 4D studying of human actions.

Our algorithm and our dataset are available in Section VII of the dataset in https://doi.org/10.4121/3a5eb5a8-bbae-4dd9-9a8d-d621bc1e36d2.v1. The algorithm was implemented using MatlabR2022a on a computing platform with an Intel Core-i5 9600 K 4.6 GHz processor.

C. DATA-SET COMPARISON

To the best of our knowledge and referring to Sections I and II, there are few articles that developed a 4D data-set based on a software-based frame time-synchroniser, while we recognised a few works with similar state-of-the-art results summarised in Table 4 as Walking Foot [48], 4DComplete [25], Dynamic foot [5], and SURP [34]. According to the table, we compare the presented data-sets in the works with the results of our work in this article using several matrices: Number of objects, Number of cameras, speed, total frames, time-delay synchronisation, and accuracy.

Briefly, the work in [48] presents a step-by-step semi-automated framework to reconstruct a full walking foot using 7 RealSense cameras, 4DComplete [25] includes animation sequences of animals and humans body, the work in [5] introduced a human feet data-set based on a parametric statistical shape model, and SURP [34] is a data-set including different human body part and full human body shapes.

According to Table 4, the work in [48] presents more accuracy than our work (due to manual filtering and time-delay synchronisation), Dynamic foot [5] shows higher speed than

TABLE 4. Comparison of Available Data-Sets

Work in	Number of objects	Number of cameras	Speed (fps)	Total frames	Time-delay synchronisation	Accuracy (cm)
Walking Foot [49]	3	7 (RealSense)	5	15	Manually	0.2
4DComplete [25]	31	1 (unknown)	unknown	1.972 k	not applicable	3.74
Dynamic foot [5]	30	6 (RealSense)	90	1.771 k	Manually	1
SURP [34]	Not available	8 (3DMD)	10	1200 k	Manually	0.9
Ours	116	6 (Azure Kinect)	15	5.147 k	Automatically	0.68

our work, and SURP [34] has more total frames than our work; however, the framework we introduced is fully automated especially fro time-delay synchronisation which is completely novel, and the data-set we presented is comparatively including more than four times objects than the other works. In addition, excluding the semi-automated work in [48], our work outperforms the other compared works in accuracy for an average of 45% which is a considerable achievement.

VII. CONCLUSION

In this article, we proposed a generic framework to synchronize and register asynchronously captured point clouds of a moving and deforming object, namely the human foot, through a novel ADGC-LSTM-based network and a non-rigid registration algorithm. We implemented the framework on the data captured from a novel 4D foot scanner to acquire the first 4D open-access feet data-set with the focuses on 1) finding the dynamic connectivity among 3D scans captured at different timestamps of each camera in terms of dynamic feature synchronization and 2) extracting meaningful dynamic features from the combined views of multiple cameras for estimating the amplitude of the deformation. Experiment results show that our method improved the synchronization process on average by about 30% compared to other state-of-the-art methods. Meanwhile, the quality of the acquired 4D scan was comparatively high regarding the deformation of each part of the foot, and such information can be useful in different applications, e.g. footwear design. Further developments include establishing a 4D Statistical Shape Model (SSM) of human foot as a tool to study the gait and foot deformations. Also, due to inconsistency in capturing speeds of different cameras, there are differences in the resolutions of the frames, which might be improved by using temporal super-resolution repetitive motion methods.

REFERENCES

[1] B. Amberg, S. Romdhani, and T. Vetter, "Optimal step nonrigid ICP algorithms for surface registration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[2] C. J. Blackwell, J. Khan, and X. Chen, "54-6: Holographic 3d telepresence system with light field 3D displays and depth cameras over a lan," in *Proc. SID Symp. Dig. Tech. Papers*, vol. 52, 2021, pp. 761–763.

[3] L. M. Boorady, "Functional clothing–Principles of fit," NISCAIR-CSIR, India., 2011, pp. 344–347.

[4] L. M. Boorady, M. Rucker, C. Haise, and S. P. Ashdown, "Protective clothing for pesticide applicators: A multimethod needs assessment," *J. Textile Apparel, Technol. Manag.*, vol. 6, no. 2, 2009.

[5] A. Boppana and A. P. Anderson, "Dynamic foot morphology explained through 4 d scanning and shape modeling," *J. Biomech.*, vol. 122, 2021, Art. no. 110465.

[6] A. Bozic, P. Palafox, M. Zollhöfer, A. Dai, J. Thies, and M. Nießner, "Neural non-rigid tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 18727–18737 .

[7] A. Bozic, M. Zollhofer, C. Theobalt, and M. Nießner, "DeepDeform: Learning non-rigid RGB-D reconstruction with semi-supervised data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7002–7012.

[8] E. Bye, K. L. Labat, and M. R. Delong, "Analysis of body measurement systems for apparel," *Clothing Textiles Res. J.*, Vol. 24, no. 2, pp. 66–79, 2006.

[9] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6299–6308.

[10] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.

[11] M. Dekker, "Mathematical Programming," Boca Raton, Fl, UsA: CRC, May 1986.

[12] C. Feichtenhofer, "X3D: Expanding architectures for efficient video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 203–213.

[13] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6202–6211.

[14] J. E. Freund, I. Miller, and M. Miller, *John E. Freund's Mathematical Statistics: With Applications*. London, U.K.: Pearson Education, 2004.

[15] A. Geiger, F. Moosmann, Ö Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 3936–3943.

[16] X. Gu, Y. Wang, C. Wu, Yong J. Lee, and P. Wang, "HPLFlowNet: Hierarchical permutohedral lattice flownet for scene flow estimation on large-scale point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3254–3263.

[17] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 922–929.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time volumetric non-rigid reconstruction," in *Proc. 14th Euro. Conf. Comput. Vis.*, 2016, pp. 362–379.

[20] T. U. Kamble and S. P. Mahajan, "3d vision using multiple structured light-based kinect depth cameras," *Int. J. Image Graph.*, 2022, Art. no. 2450001.

[21] P. Diederik, Kingma, and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[22] F. Kwa, "Design of an accurate and low cost 4D foot scanner for podiatrists," M.S. thesis, Delft Univ. Technol., Delft, Netherlands, 2021.

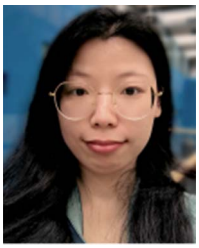
[23] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-structural graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3595–3603.

[24] Y. Li, A. Bozic, T. Zhang, Y. Ji, T. Harada, and M. Nießner, "Learning to optimize non-rigid tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4910–4918.

[25] Y. Li, H. Takehara, T. Taketomi, B. Zheng, and M. Nießner, "4dComplete: Non-rigid motion estimation beyond the observable surface," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 12706–12716.
[26] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "GeoMAN: Multi-level attention networks for geo-sensory time series prediction," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 3428–3434.
[27] J. Liu and D. Xu, "GeometryMotion-Net: A strong two-stream Baseline for 3d action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 12, pp. 4711–4721, Dec. 2021.
[28] X. Liu, M. Yan, and J. Bohg, "MeteorNet: Deep learning on dynamic 3D point cloud sequences," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9246–9255.
[29] L. Ma, X. Li, J. Liao, and P. V. Sander, "3d video loops from a. input," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 310–320.
[30] Microsoft, "Synchronize multiple azure kinect DK devices," 2020. [Online]. Available: https://learn.microsoft.com/en-us/azure/kinect-dk/multi-camera-sync
[31] T. Morimoto and I. Mitsugami, "Motion capture system by spatio-temporal integration of multiple kinects," in *Proc. IEEE 8th Glob. Conf. Consum. Electron.*, 2019, pp. 1158–1159.
[32] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 343–352.
[33] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4d normals for activity recognition from depth sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 716–723.
[34] A. A. A. Osman, T. Bolkart, D. Tzionas, and M. J. Black, "SUPR: A sparse unified part-based human body model," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 568–585.
[35] D. Pickup et al., "SHREC'14 track: Shape retrieval of non-rigid 3D human models," in *Proc. 7th Eurographics Workshop 3D Object Retrieval*, 2014, pp. 1–10.
[36] R. Poppe, "A survey on vision-based human action recognition," *Image Vis. Comput.*, vol. 28, no. 6, pp. 976–990, 2010.
[37] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.
[38] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet : Deep hierarchical feature learning on point sets in a metric space," *Adv. Neural Inf. Process. Syst.*, 2017, vol. 30, pp. 1–10.
[39] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with directed graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7912–7921.
[40] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional lstm network for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1227–1236.
[41] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic, "KillingFusion: Non-rigid 3d reconstruction without correspondences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1386–1395, 2017.
[42] J.-J. E. Slotine et al., *Applied Nonlinear Control*, vol. 199, Englewood Cliffs, NJ, USA: Prentice Hall, 1991.
[43] O. Sorkine and M. Alexa, "As-rigid-as-possible surface modeling," in *Proc. Symp. Geometry Process.*, 2007, pp. 109–116.
[44] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, "Human action recognition from various data modalities: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3200–3225, Mar. 2023.
[45] F. Tajdari, "Advancing non-rigid 3 d/4d human mesh registration for ultra-personalization," Ph.D. dissertation, Delft Univ. Technol., Delft, Netherlands, 2023.
[46] F. Tajdari, T. Huysmans, and Y. Song, "Non-rigid registration via intelligent adaptive feedback control," *IEEE Trans. Visual. Comput. Graph.*, early access, Jun. 08, 2023, doi: 10.1109/TVCG.2023.3283990.
[47] F. Tajdari, T. Huysmans, Y. Yang, and Y. Song, "Feature preserving non-rigid iterative weighted closest point and semi-curvature registration," *IEEE Trans. Image Process.*, vol. 31, pp. 1841–1856, 2022.
[48] F. Tajdari, F. Kwa, C. Versteegh, T. Huysmans, and Y. Song, "Dynamic 3d mesh reconstruction based on nonrigid iterative closest-farthest points registration," in *Proc. Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, vol. 2022, pp. 1–9.
[49] M. Tajdari et al., "Image-based modelling for adolescent idiopathic scoliosis: Mechanistic machine learning analysis and prediction," *Comput. Methods Appl. Mechanics Eng.*, vol. 374, 2021, Art. no. 113590.
[50] U. H. Tang, J. Siegenthaler, K. Hagberg, J. Karlsson, and R. Tranberg, "Foot anthropometrics in individuals with diabetes compared with the general swedish population: Implications for shoe design," *Foot Ankle Online J*, vol. 10, no. 3, p. 1, 2017.
[51] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.
[52] S. Valette, Jean M. Chassery, and R. Prost, "Generic remeshing of 3D triangular meshes with metric-dependent discrete Voronoi diagrams," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 2, pp. 369–381, Mar./Apr. 2008.
[53] A. Visioli, *Practical PID Control*. Berlin, Germany: Springer Science & Business Media, 2006.
[54] L. Wang et al., "Temporal segment networks: Towards good practices for deep a. recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36.
[55] P. Wang, W. Li, Z. Gao, C. Tang, and P. O. Ogunbona, "Depth pooling based large-scale 3D action recognition with convolutional neural networks," *IEEE Trans. Multimedia*, vol. 20, no. 5, pp. 1051–1061, May 2018.
[56] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, "RGB-D-based human motion recognition with deep learning: A survey," *Comput. Vis. Image Understanding*, vol. 171, pp. 118–139, 2018.
[57] X. Wang, Y. Yan, H-M. Hu, B. Li, and H. Wang, "Cross-modal contrastive learning network for few-shot action recognition," *IEEE Trans. Image Process.*, vol. 33, pp. 1257–1271, 2024.
[58] Y. Wang et al., "3DV: 3D dynamic voxel for action recognition in depth video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 511–520.
[59] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, 2003.
[60] A. Xiao, J. Huang, D. Guan, X. Zhang, S. Lu, and L. Shao, "Unsupervised point cloud representation learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 11321–11339, Sep. 2023.
[61] Y. Xiao et al., "Action recognition for depth video using multi-view dynamic images," *Inf. Sci.*, vol. 480, pp. 287–304, 2019.
[62] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. Thirty-second AAAI Conf. Artif. Intell.*, 2018.
[63] X Yang and Y. Tian, "Super normal vector for human activity recognition with depth cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 1028–1039, May 2017.
[64] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1963–1978, Aug. 2019.
[65] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," *Model. Financial Time Ser. S-PLUS*, pp. 385–429, 2006.
[66] M. Zollhöfer et al., "Real-time non-rigid reconstruction using an rgb-d camera," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 1–12, 2014.

FARZAM TAJDARI received the Ph.D. degree from the School of Engineering, Aalto University, Espoo, Finland, in 2023, and the second Ph.D. degree in mechatronic design engineering from the Delft University of Technology, Delft, The Netherlands, in 2023. Since 2024, he has been a Postdoc Researcher with the Faculty of Mechanical Engineering, Delft University of Technology. In 2023, he was a Postdoc with ME department, the Eindhoven University of Technology, Eindhoven, The Netherlands, where he is currently a Guest Postdoctoral Researcher. His research interests include encompass control, and non-linear systems, addressing challenges in the fields of ITS, privacy of dynamic systems, and geometry processing.

TOON HUYSMANS received the Ph.D. degree from the Department of Physics, University of Antwerp, Antwerp, Belgium. He is currently an Assistant Professor of digital human modelling with the Faculty of Industrial Design Engineering, Delft University of Technology, Delft, The Netherlands. His main research interests include anthropometric and biomechanical digital human modelling and simulation with a focus on data-driven methods, and design tools in the fields of ergonomics, anatomy, and orthopaedics.



XINHE YAO received the Ph.D. degree from the Faculty of Industrial Design Engineering, Delft University of Technology, Delft, Netherlands, in 2023. She is currently working as a Postdoctoral Fellow with the Industrial Design Institute, COMAC Shanghai Aircraft Custumer Service Company, Ltd. Her research interests include passenger comfort and physical ergonomics in aircraft cabins.

JUN XU received the M.S. degree in biomedical engineering in 2018 from Shanghai University, Shanghai, China, where he started his Doctor Program. He is currently working toward the Ph.D. degree with the Delft University of Technology, Delft, The Netherlands. His research focuses on 3D printed electronics.

MARYAM ZEBARJADI (Graduate Student Member, IEEE) received the M.S. degree in electrical engineering from the University of Tehran, Tehran, Iran. She is currently working toward the Ph.D. degree with the electrical Engineering department, University of Minnesota, Minneapolis, MN, USA. Her research interests include in the field of computer vision, signal and image processing, machine learning, and healthcare analytics.

YU SONG (Member, IEEE) received the Ph.D. degree from the Department of Mechanical Engineering, The University of Hong Kong, Hong Kong. He joined the Faculty of Industrial Design Engineering, Delft University of Technology, Delft, The Netherlands, in 2001. He is currently an Associate Professor with the Department of Sustainable Design Engineering. His main research interests include human digital twin, 3D scanning, and Ergonomics.