IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 2, 2022, accepted February 28, 2022, date of publication March 8, 2022, date of current version March 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3157939

Scenario-Based Sensed Human Motion Editing and Validation Through the Motion-Sphere

BHARATESH CHAKRAVARTHI, ASHOK KUMAR PATIL, JAE YEONG RYU, ADITHYA BALASUBRAMANYAM, AND YOUNG HO CHAI

Virtual Environments Laboratory, Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Young Ho Chai (yhchai@cau.ac.kr)

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government Ministry of Science and ICT (MSIT) (No. 2018-0-00599, Proxemics based Pervasive Interactions for the Wide area·High speed·Serial Motion Recognition).

ABSTRACT Synthesizing realistic human motion data using a real-time motion capture system in a controlled environment is a critical challenge. In addition, effectively manipulating the existing motion data is another primary concern and using such modified data in human motion analysis and activity recognition systems are prone to errors. This paper presents a simplistic and comprehensive system to effortlessly author, edit, and validate human motion data. The system enables a naive user to edit the existing motion data interactively using a humanoid model in a three-dimensional space, based on user-defined scenarios and synthesize numerous variations of the motion sequences. A modular concept of scenario-based sensed unit motion editing has been adopted to demonstrate the proposed system. We employed an efficient analytical kinematic and constraint solver to enforce the inherent body joint limits and external constraints while editing to synthesize complete and meaningful motion sequences. Furthermore, we substantiated the proposed sensed unit motion editing framework through a visual validation study using an open-source intuitive visualization tool called the Motion-Sphere. Finally, we compared the resultant synthesized motion against the real-time motion capture system data to verify the body segments' orientation and position accuracy deviations.

INDEX TERMS Data visualization, motion authoring, motion capture system, motion editing, motion reconstruction, motion-sphere, kinematics, and constraints.

I. INTRODUCTION

Recent advances in the field of inertial measurement unit (IMU) based motion capture (MoCap) systems have opened new avenues for human motion retargeting, activity recognition, movement analysis, and motion data visualization. A wide range of real-time MoCap systems and three-dimensional (3D) human-like character animation systems are available for synthesizing realistic human motion data. Computer graphics professionals and researchers have used distinctive methods to construct numerous subject-specific motion databases to benefit the human-computer interaction researchers and the virtual reality user community [1], [2].

Despite the availability of MoCap systems and animation software, human motion data acquisition and its practical usage remains challenging. The IMU-based MoCap systems (e.g., Xsens, Perception Neuron, and Vicon) used to track and record human actions require a controlled environment, pre and post processing operations, and countermeasures to reduce sensor drift errors [3], [4]. The 3D character animation software (e.g., 3ds Max and MotionBuilder) used to synthesize human motion, requires proficient skills and tedious efforts. Further, the platform dependency and lack of customization increase the complexity, making the process difficult from an end-user perspective [5]. Human motion data centred research greatly relies on preexisting motion databases (e.g., CMU MoBo and KIT Whole-Body) [6]–[8]. However, these databases lack a standard controlled methodology required for the quantitative evaluation of versatile problems. In addition, motion databases are subjected to several factors, such as the diverse nature of MoCap systems, data sampling methods, range of motions, and discrete data file formats. Therefore, every user needs to select their data sets of interest individually and invest considerable effort into

The associate editor coordinating the review of this manuscript and approving it for publication was Walter Didimo.

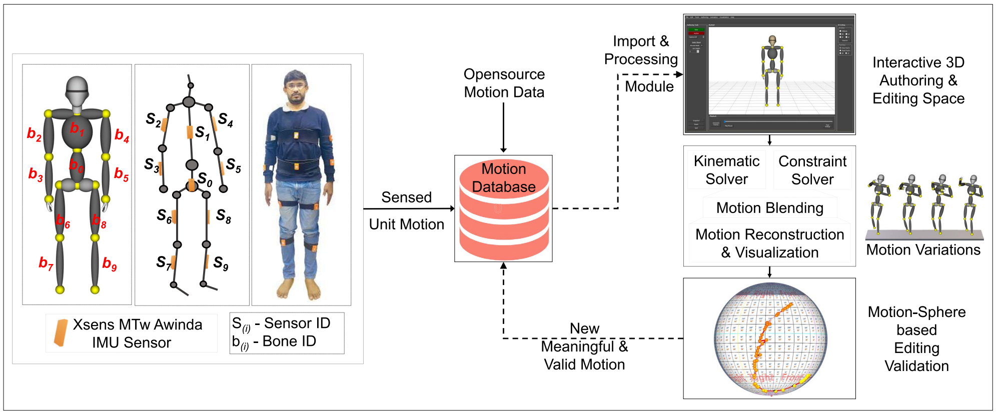


FIGURE 1. Overview of the proposed interactive human motion editing system to synthesize numerous variations of the existing motion data and automatic editing validation using Motion-Sphere.

processing the data to be used [9], [10]. The applications of human activity recognition and human motion analysis require 3D articulated figures, humanoid models, and virtual avatars. Human motion reconstruction using these articulated models is subject to a large number of body joint degree of freedom (DoF). In addition, it is crucial to address various types of constraints such as, the inherent body joint rotation limit, body segments overlapping, and external entity contact. Overcoming these shortcomings during motion data acquisition remains a primary concern, and these factors limit the reusability of the existing motion data. Therefore, it is necessary to efficiently utilize the existing human motion data and manipulate it without losing human movements' naturalness.

This paper presents a modular open-source platform to interactively author and edit full-body human motion data in a 3D space. First, we record a basic set of human motion sequences, called the sensed unit motions using an IMU-based motion capture system (Xsens MTw Awinda) in real-time. The sensed motion data segments are collectively stored as a motion database. Then, the user can import and visualize motion data using a humanoid model in an interactive 3D space, where the proposed authoring and editing system automatically maps the sensed motion data to the respective bone segments of the humanoid model. Next, the users can specifically choose the motion sequences of their interest and edit further based on different scenarios to synthesize an entirely new set of meaningful motion sequences based on predefined scenarios. Finally, the edited motion can be visually validated using an open-source visualization tool called Motion-Sphere [11]. Figure 1 depicts the overview of the proposed human motion editing and validation system. The key contributions of the proposed interactive application system are summarized below:

- A seamless interactive interface to edit and modify the existing MoCap data.
- Minimal sensed unit motion collection to synthesize a new set of meaningful motion sequences.
- Human body kinematics incorporated motion editing solver with spatio-temporal constraints.
- An intuitive method to visually validate the edited motion data using Motion-Sphere.
- Motion data composition from scratch, in the absence of MoCap systems.

We introduce the concept of minimal sensed unit motion collection and demonstrate the superiority of the proposed 3D human motion editing and authoring system with scenario-based reactive motion synthesis under different spatio-temporal constraints using an analytical kinematic and constraint solver. Furthermore, we consider the human body kinematics as a critical parameter to verify the correctness of the resultant edited motion. In this regard, we validate the edited motion's kinematic correctness and plausibility using Motion-Sphere. Further, we evaluated the naturalness of the newly synthesized motion sequence, against MoCap captured data as the ground truth to determine the deviations in bone orientations incurred during the editing process. The proposed system serves as a comprehensive toolkit to synthesize numerous realistic variations of the human motion data and improves the reusability of the existing motion databases via simplistic editing operations. In addition, the proposed system serves as a solution to an ever-increasing demand for human motion data in various application domains, such as 3D character animation, computer vision, robotics, fitness, sports training, and rehabilitation.

The rest of the paper is organized as follows: The various human motion editing principles and practices are discussed in Section II. The proposed sensed motion editing system design is presented in Section III, followed by the motion editing validation study in Section IV. Next, use-case assessment, comparative study and discussion are presented in Section V, followed by the conclusion in Section VI.

II. BACKGROUND

Human activity tracking and motion data acquisition are widely required in computer animation, human-computer interaction, and virtual reality. With the surge in MoCap technology, several research studies have been conducted to formulate all the necessary processes to reckon human body dynamics and kinematics to synthesize human motion. In addition, numerous investigations have been performed to develop feasible methods for editing and reusing the existing human motion data. Moreover, numerous 3D character animation application systems are reported to produce human-like character animation.

Wang *et al.* [12] performed a comparative study to outline the different motion synthesis techniques used by researchers, and animation professionals. In contemporary research, the IMU-based MoCap system data-driven methods are extensively used. Computer graphics and animation practitioners employ human-like character animation software such as, Autodesk Maya, 3ds Max, MotionBuilder, Adobe Animate, and Blender, to synthesize motion data artificially in different types of 3D file formats (*.mp, *.3ds, *.fbx, *.blend, and *.bvh) [13], [14]. Furthermore, wearable and ambient sensing technology based systems are commonly used by the motion capture system research community. Retroreflective or light-emitting diode markers-based optical sensors, video/marker-less tracking methods, and IMU-based wearable sensors are used to record human actions and store them as motion dataset. Currently, several human motions databases [1], [8], [15] are available that offer motion data in different formats (*.trc, *.htr, *.amc/asf, *.mvn, and *.c3d) [16], [17]. The 3D human motion data constitutes translatory and rotational motion related to the frontal, sagittal, and transverse planes. In practice, the motion data is either acquired using MoCap systems in real time or artificially synthesized using different application software. After that, to utilize the synthesized motion data efficiently, it requires preferred intended motion editing operations [18]. Inverse Kinematics (IK) principles serve as the best means to edit full-body human motion by enforcing user-defined constraints. Aristidou *et al.* [19] presented the most popular IK methods in terms of performance, computational cost, and smoothness. Gleicher *et al.* [20] compared various constraints-based editing methods and evaluated their suitability for online and offline editing applications. The most efficient techniques in the taxonomy of motion editing [20] are per-key methods, motion warping and displacement mapping, per-frame methods, and space-time methods.

To date, many space-time domain-based motion editing techniques have been developed using motion warping techniques to manipulate the input motion sequences while overcoming specified constraints. A space-time constraint solver-based editing technique [21] considers the entire motion in making changes simultaneously. Contrarily, Lee *et al.* [22] presented a multilevel B-spline fitting technique-based editing method by considering the inter and intra-frame relationships.

Tak *et al.* [23] proposed a per-frame Kalman filter framework to ensure the kinematic and dynamic accuracy and overcome the overhead computational cost in the space-time optimization techniques. Shin *et al.* [24] proposed a dimensionality reduction technique based motion editing in low dimensional space where the high dimensional pre captured motion data are represented as streams of curves.

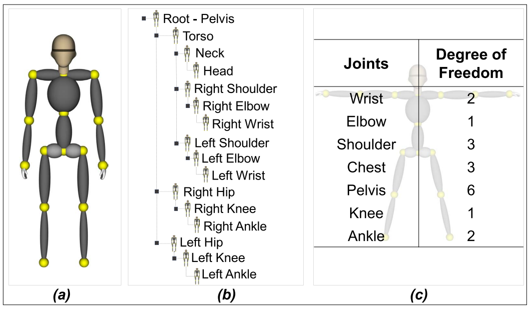
A naive user needs to invest substantial effort into understanding the complex editing principles. Unfortunately, the existing motion editing systems lack an interactive means to author and edit MoCap data irrespective of the motion data file format and differences in the underlying human motion data acquisition system. Also, there exists no means to visually validate the kinematic correctness of the resultant edited motion data.

III. SENSED MOTION EDITING SYSTEM DESIGN

The proposed system enables users to import and edit motion data from our unit motion collection and the existing open-source databases [25]–[27]. The import and preprocessing components of the editing module consists of open-source APIs to parse the motion data files (*.fbx, *.bvh, and *.txt). Users can perform editing operations using the proposed end effector-based analytical kinematic solver and enforce a defined set of inherent body joint limits and external constraints to compose user-defined scenario-based motion sequences. The sensed motion data can be edited independently and then combined to form complete and meaningful new motion sequences. Subsequently, the new motion sequences can be reconstructed and visualized on humanoid models and recursively perform editing operations to compose the desired motion sequences. Users can perform several editing operations interactively at the per-frame, inter-frame, intra-frame, and full-frame levels. Further, a user can also synthesize the motion data without using a MoCap system in a custom environment, with the aid of a hierarchical kinematic humanoid model. The user is provided all the necessary tools to synthesize the motion data in a user-friendly way. After completing the motion composition task, the user can export the motion data in terms of the positions and orientations (quaternion form) in a textual file format [28]. The retrieved motion data can be reconstructed and visualized on different humanoid models. Thus, the proposed system serves two purposes – to author new motion from scratch in the absence of the MoCap system and to edit existing sensed motion data interactively.

A. EDITING MODULE

The sensed motion editing system is a graphical user interface-based interactive framework developed using an open-source software. A hierarchical humanoid model is used to map the motion data and edit the intended motion sequence through a series of mouse-based user interactions. The editing module enables the user to synthesize new 3D human motions consistent with the inherent body joint constraints. Our system allows forward kinematics (FK) and IK operations based on specific pose compositions. Users can switch between the FK and IK modes to compose different poses. Figure 2(a) depicts a humanoid model in the attention pose, Figure 2(b)



Joints	Degree of Freedom
Wrist	2
Elbow	1
Shoulder	3
Chest	3
Pelvis	6
Knee	1
Ankle	2

FIGURE 2. Humanoid model used for authoring and editing: (a) attention pose, (b) body segments hierarchy, and (c) body joints DoF.

shows the hierarchical structure of the humanoid model, and Figure 2(c) shows the default configuration for each body joint DoF with a 14-segment, 18-DoF model as an instance. Users can change and reconfigure the body joints' DoF based on the target motion plan.

The humanoid model has a kinematically linked hierarchy with the pelvis segment as the root node. Users can scale the model to different heights and author different types of motion sequences. In addition, the editing module offers fixed foot and free foot modes to compose static and transforming motions. The chest, neck, upper arm, and upper leg segments inherit their movement relatively from the pelvis segment. The lower arm and lower leg segments inherit movements from their immediate parent segments. The hand and foot segments are the child segments of the lower arm and lower legs, respectively.

With the IK mode in use, hand and foot segments are considered as end effectors to synthesize different poses with an automatic orientation update for elbow and knee segments. Each body joint segment will possess a local axis of rotation and DoF with which the user can make the model move naturally. For example, the chest forward bend motion will result in automatic arm movement with hands touching the ground. With a reference motion scheme, the user can select and rotate joint segments with a restricted DoF along their local axis to author key poses and capture the associated data frame at different time intervals. A captured keyframe comprises individual segment transformations in the form of unit quaternions *(w,x,y,z)*. Next, the full-body motion data are estimated by applying spherical linear interpolation (SLERP) on the quaternion keyframes using Equation 1.

$$q = \frac{q_a * sin((1-t)*\theta) + q_b * sin(t*\theta)}{sin(\theta)} \quad (1)$$

where, 'q' is the interpolated quaternion, 'q_a' is the first quaternion keyframe, 'q_b' is the second quaternion keyframe, 't' is a scalar between 0.0 at 'q_a' and 1.0 at 'q_b', and 'θ' is half the angle between 'q_a' and 'q_b' and it is defined as,

$$\theta = acos((q_a w * q_b w) + (q_a x * q_b x) + (q_a y * q_b y) + (q_a z * q_b z)) \quad (2)$$

Algorithm 1: Steps to Configure and Edit Motion Data

Initialize: Humanoid model
B_s**:** A set of bone segments $(0 \ldots n-1)$
J_l**:** A set of joint limits $(DoF's)$
C**:** A set of constraints $(Spatio - temporal)$
Result: A new motion sequence file
while *Bone segments are manipulated* **do**
  for $B_s = 0, 1, \ldots (n\text{-}1)$ **do**
    $M_e := 0$
    foreach B_s *transformation* **do**
      $J_l \leftarrow KinematicSolver(T, B_s)$;
      $C \leftarrow ConstraintSolver(S, T)$;
      $kf \leftarrow kf_n$;
    end for
    for $kf_{(0)}$ *to* $kf_{(n)}$ **do**
      $Q_{Interpolation}(sQ_a, Q_b, t)$
    end for
  end for

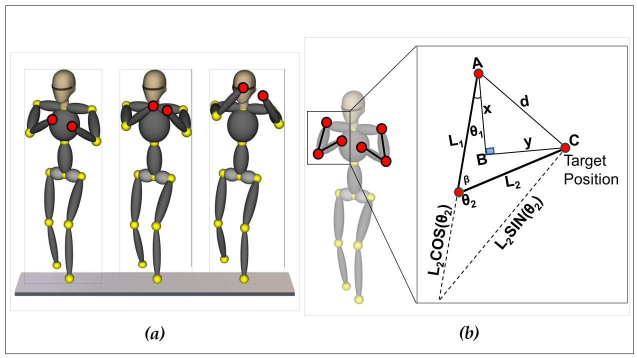


FIGURE 3. End effectors-based kinematics editing configuration: (a) shows right and left-hand segments as the end effectors and their change in position, and (b) IK configuration.

The motion editing phase starts by loading existing motion data and mapping it with a humanoid model of choice. Then, configure all the parameters required to edit and compose poses. After that, SLERP is applied between the edited keyframes at a defined interval of time.

Algorithm 1 shows the steps for editing motion sequences and exporting resultant motion data in a file. To synthesize fast, medium, or slow-motion sequences, users can define the number of intermediate frames between two static key poses. The feedback-based analytical IK solver employed in the authoring module enables the user to synthesize kinematically valid motions with inherent body joint constraint. The hand and foot segments can be parameterized to reach specific target positions obeying the joint rotation constraints. The body joint-segment DoF constraint can be defined and restricted by the user inputs before starting the motion composition phase.

To minimize the computational cost and recursively solve an IK problem based on scenarios while composing poses,

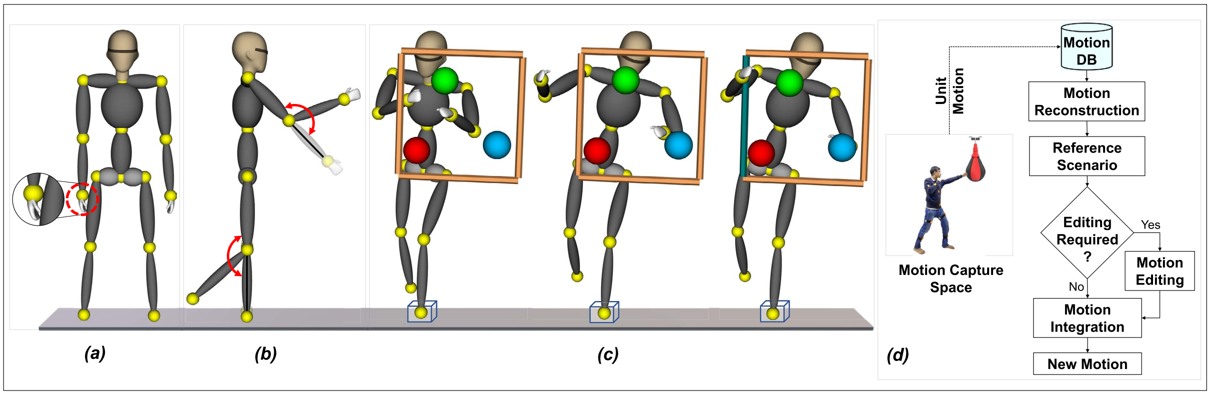


FIGURE 4. Different types of constraints enforced for scenario-based editing: (a) body segments contact constraint, (b) inherent body joint rotation constraint, (c) external contact constraint, and (d) steps to synthesize new motion sequences based on scenarios.

Algorithm 2: Analytical IK Solver Algorithm

Input: L_1, L_2, x, y
Output: θ_1, θ_2
Step 1: Configure joint rotation limits
Step 2: Move end-effectors to target position (*Mouse Event*)
if *ConstraintViolation is true* **then**
  Invalid solution;
  return *closest alternative*;
else
  return *Valid solution*
Compute θ_2,

$$\theta_2 = arccos\frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}$$

Compute θ_1,

$$\theta_1 = atan2(\dot{y}, \dot{x})$$

where

$$\dot{x} = x(L_1 + L_2cos\theta_2) + y(L_2sin\theta_2)$$
$$\dot{y} = y(L_1 + L_2cos\theta_2) - x(L_2sin\theta_2)$$

Repeat *Steps*, till end-effector reaches target position

an analytical IK solver with reduced complexity is employed in this work. To compose a pose using an IK solver, user can use the end effectors (i.e., right hand, left hand, right foot, and left foot segments) and author target or goal-based poses in 3D space. Figure 3(a) depicts right and left arm movement poses using the end-effector based analytical IK solver used in this system, where L_1 and L_2 indicate the length of upper and lower arm segment respectively, and θ_1 and θ_2 indicate upper arm joint and elbow joint angles. Algorithm 2 depicts the steps and equations used to solve θ_1 and θ_2 as in Figure 3(b). User-defined joint movement constraints can be enforced through the constraints configuration module of the editing system. The joint rotation limit constraints offered by the analytical IK solver enables the user to author natural poses similar to human body anatomy. In addition, the oriented bounding box (OBB) tree-based collision detection filter [29] is used to implement body segment occlusion, foot on ground, and external entity contact constraint.

Figure 4 shows the humanoid model with different constraints configured. Figure 4(a) depicts the body segments overlapping constraint where the right hand segment comes in contact with the right upper leg segment and constraint violation is reflected by restricting further movement such that no two body segments cross over. Figure 4(b) depicts inherent body joint rotation constraints applied for arm and leg segment's flexion and extension movements respectively. Figure 4(c) shows target hitting scenario with vertical bars as external obstacle constraints. Whenever any of the body segment comes in contact with vertical bars, the external contact constraint violation is indicated with a change in color of vertical bars and stops further cross-over movements.

B. SENSED UNIT MOTION CONCEPTUALIZATION

The motion editing framework presented in this section serves as an intuitive tool to synthesize numerous variations of existing sensed human motion. We used Xsens motion tracking system [30] to capture the basic set of subject specific poses to create smaller motion sequences called unit motions. Unit motion constitutes 'n' number of motion frames, where each frame corresponds to the position and orientation of body segments represented as unit quaternions. Human motion data is characterized by keyframes depicting a static pose, and unit motion is a sequence of motion between any two given keyframes. The idea is to use minimal set of unit motions and edit them further selectively to synthesize numerous variations, and blend different combinations based on predefined scenarios to create completely new and meaningful motion sequences.

The unit motions are edited considering the joint rotations in terms of swing and twist, target position-based pose composition, pose duplication, adding or removing motion chunks, speed of the motion (by increasing and decreasing

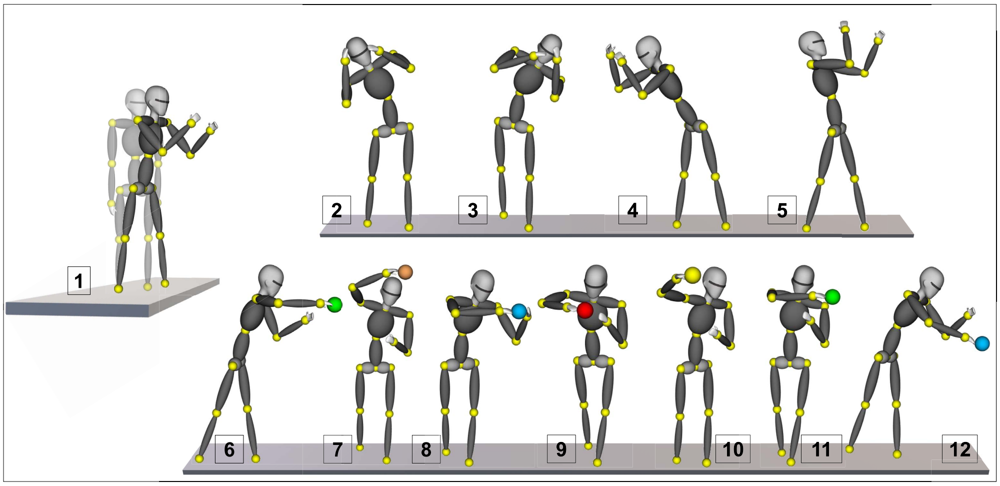


FIGURE 5. The 12 primary sensed boxing hook motion sequences: (1) attention pose to hook ready pose, (2-5) chest segment abduction, adduction, flexion, and extension poses, (6-12) right arm hook poses to hit targets placed at different positions with respect to body's mid-line.

the number of frames), and applying different types of constraints. Figure 4(d) shows the sequence of steps to synthesize new motion by editing unit motion selected from the sensed unit motion database. Users can select the unit motion from the data collection to perform edit and blend operations to create a meaningful motion sequence. The resultant edited motion sequences are stored along with unit motions, thus increasing the number of motion data files. The proposed editing module has APIs to map and use motion data in different file formats such as.bvh and.fbx files.

C. SCENARIO-BASED MOTION EDITING

The human motion data available in several open-source databases are subjective, and reusing them effectively is a challenging task. In this section, we demonstrate a practical course of action to reuse the MoCap data to synthesize variations of the existing motion sequences. We captured 12 primary boxing hook motion sequences from the hook-ready pose to different target hitting poses, as shown in Figure 5. The sensed unit motion collection is comprised of a hook ready pose (pose 1) from attention pose, chest segment flexion, extension, abduction, and adduction motion sequences (pose 2 to pose 5) with both the hands in the hook-ready pose and seven different targets hitting (using right arm) motion sequences (pose 6 to pose 12) are captured. The targets were placed at different positions (right, left, up, down, long, middle, short-range) from the mid-line of the body. These 12 primary motion sequences are further edited independently based on specific scenarios to synthesize new motion sequences.

Figure 6(a) is an example scenario wherein a single-tap hook motion is edited to synthesize a double-tap hook motion. The analytical kinematic solver employed in this work adopts the joint anatomical constraints. The external contact constraints are enforced automatically when an end effector segment comes in contact with an external entity. In this example, when the right hand comes in contact with the target (sphere) placed at a distant position, further movement of the right arm is restricted. Trim and join, and chunk size parameters

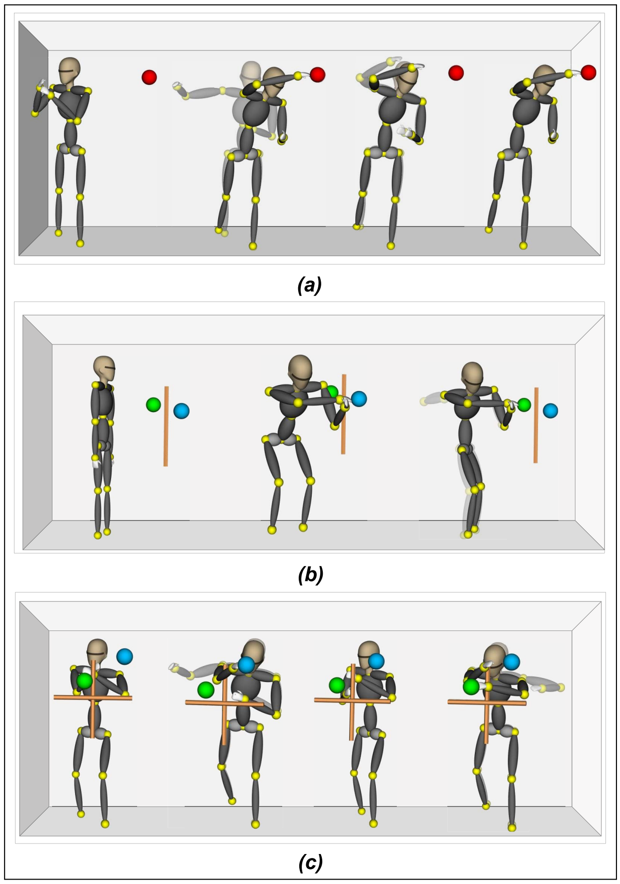


FIGURE 6. Three boxing hook scenarios: (a) double-tap hook, (b) single constraint and multi target hitting, and (c) multi-target and multi-constraint scenario.

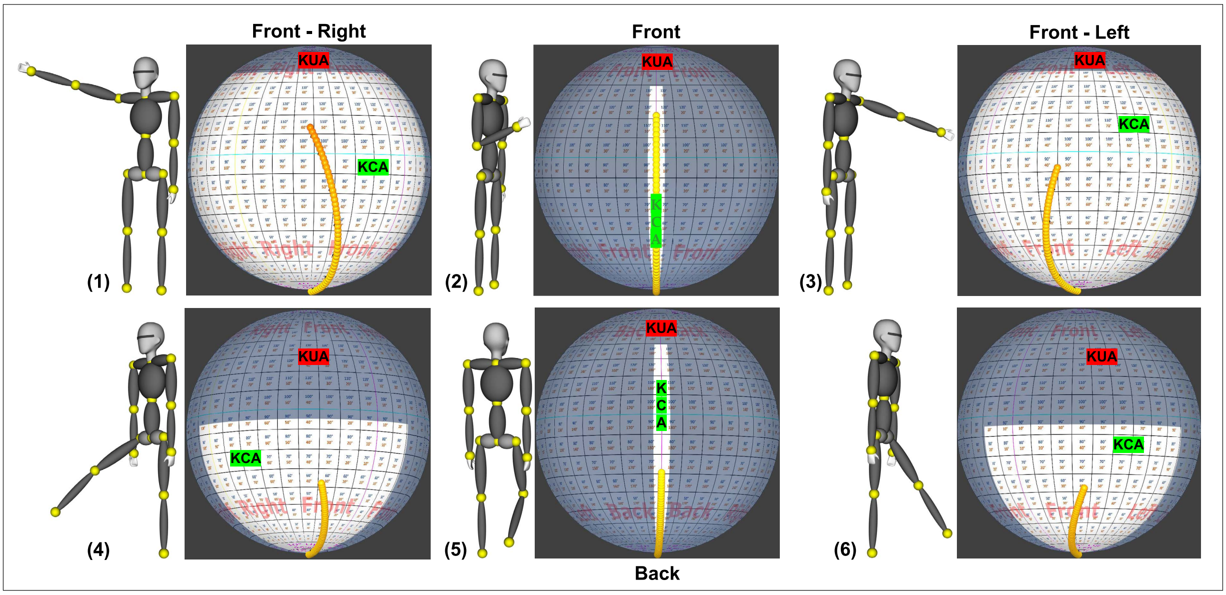


FIGURE 7. Motion-Sphere based automatic editing validation: (1 and 4) right upper-arm and upper-leg motion represented as trajectory on the Motion-Sphere's front-right KCA, (2) right lower-arm motion visualization on Motion-Sphere's front KCA, (3 and 6) left upper-arm and upper-leg motion visualization on Motion-Sphere's front-left KCA, (5) right lower-leg motion visualization on Motion-Sphere's back side.

are used to integrate a pullback and reach the target position motion frames, along with the original motion sequence, in a user input-controlled environment.

The second example of editing is subjected to a hit and react scenario with two different target points (spheres) and an external constraint (vertical bar). The targets are placed at a distant position, and the sensed unit motion is edited to reach targets without violating the external entity contact constraint (i,e., both the arm segments do not cross over the vertical bar while reaching the targets). As shown in Figure 6(b), the sequence of action to reach the targets is realized using the end effectors by tweaking and updating the shoulder joint and wrist joint angles. In addition, the chest segment's horizontal adduction is fine-tuned to balance and help the end-effector reach the target.

The third example scenario has two reachable targets with two external contact constraints (a vertical bar and a horizontal bar) as illustrated in Figure 6(c). In this example, the right arm motion is kinematically edited to reach a target (blue sphere) without violating the bar crossover constraints. Then, the sensed right arm motion is duplicated and applied to the left arm segment to reach another target (green sphere).

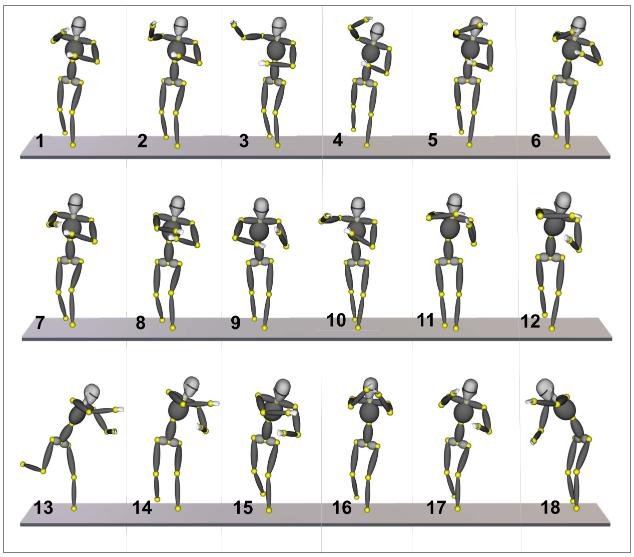


FIGURE 8. Example of kinematically valid boxing motion variations: (1-6) right arm upper hook variations, (7-18) right arm short range punch variations, and (13-18) leg segments kinematics editing variations.

IV. MOTION-SPHERE BASED EDITING VALIDATION

The modular editing system is a graphical user interface-based application developed using Qt Designer 5.12.3 and visualization toolkit (VTK), which is an open-source software system for implementing 3D graphics, modeling, rendering, and visualization. Three different types of humanoid models (male biped, female biped, and skeletal biped) are used to author and edit the human motion in 3D space. The users can perform mouse-based interaction to select and modify the body segments to compose different poses. The end-effector segments can be used to perform drag and move operations such that the model reach a target position. Autodesk FBX API and BVH parser are used to import and edit the existing motion data from open-source motion databases. The users can control and regulate authoring and editing parameters through drop-down menus and lists offered by the application system.

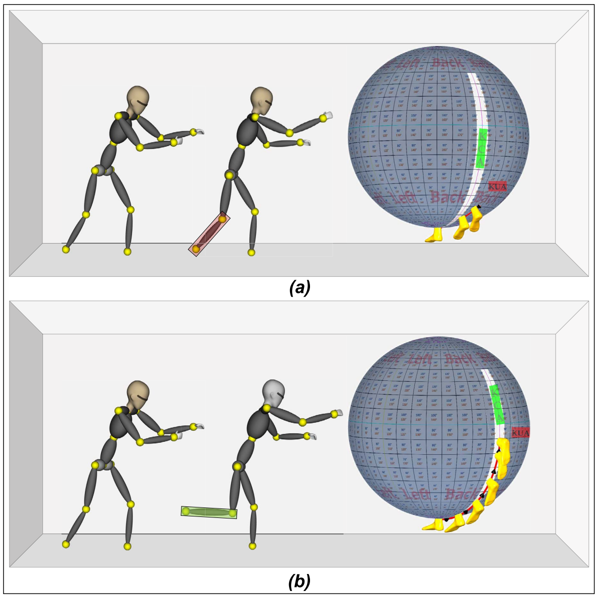


FIGURE 9. Motion-Sphere based editing validation: (a) before editing - right lower-leg sensed unit motion appearing on the KUA of Motion-Sphere depicting the erroneous induced during data acquisition, and (b) after editing - right lower-leg edited motion appearing on the KCA of Motion-Sphere.

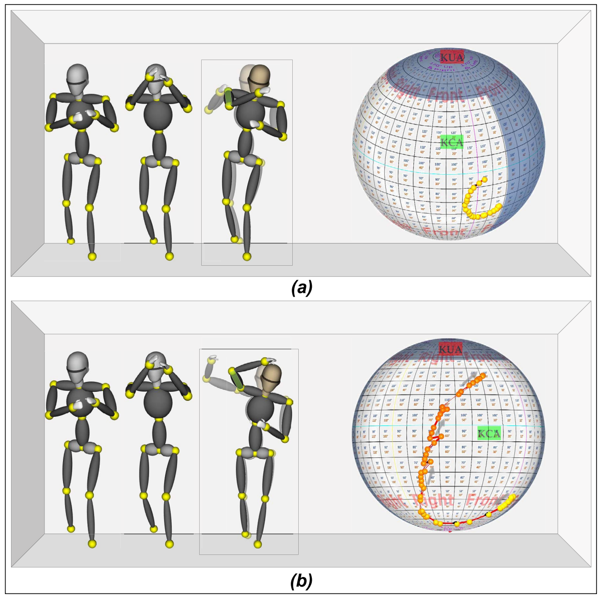


FIGURE 10. Motion-Sphere based editing validation: (a) right-arm hook motion authored and edited for a target hitting scenario and its validation, and (b) a double-tap hook motion editing validation using Motion-Sphere.

Several variations of the boxing hook motion are synthesized using a small collection of sensed unit motions, as shown in Figure 8. The sensed unit motions are selectively chosen based on specific target hitting scenarios and edited further to synthesize a new set of hook boxing motion sequences. The unit motion sequences are captured from a range of 60-120 frames per second based on the type of motion. The full-body sensed motion sequences are composed of data segments mapped to 14 skeletal joints with 18 DoF. The end-effectors are modified using the kinematics solver to make the body segments reach specific and desired target positions.

In this section, we present an experimental study to validate the motion editing system. The kinematic correctness and plausibility of the edited motions were compared and validated using the Motion-Sphere [31]. The Motion-Sphere is an intuitive tool used to visualize subtle movements of the human joints on a 3D unit sphere. Human motion visualization using Motion-Sphere aids to quantify the body joint movements and visually perform a comparative analysis. In addition, Motion-Sphere enables the user to visualize IMU sensor acquired human motion data as a trajectory and confirm the plausibility and kinematics correctness. Motion-Sphere offers kinematics constrained area (KCA) and kinematics unconstrained area (KUA) features to determine the kinematic correctness as shown in Figure 7. The KCA and KUA are represented by the highlighted and darkened areas of the Motion-Sphere. The trajectories representing the motion for a given bone segment should strictly be within the KCA. If the trajectories representing the motion of rotation constrained bone segment appear in KUA, this indicates the violation of joint DoF and its rotation constraint.

Figure 9 shows a right arm hook motion sequence. The end effector (right lower-leg) position is edited to make it reach a target position. The sphere as in Figure 9(a) represents the motion data as a trajectory appearing in the KCA for right lower leg. The rightmost sphere represents the motion data as a trajectory appearing in the KCA for right lower arm. The maximum rotation for right lower arm is constrained and the trajectory is well outside the KCA of Motion-Sphere. It can also be noticed that since it is a backward lift motion sequence, the trajectory are in the back side of the Motion-Sphere. The motion is edited to correct the lower leg kinematics and the resultant trajectory appears in the KCA of motion sphere as shown in Figure 9(b).

Figure 10 shows a right arm hook motion sequence. The end effector (right hand) position is edited to make it reach a target position. The sphere as in Figure 10(a) represents the motion data as a trajectory appearing in the KCA for right upper arm. The Motion-Sphere shows the motion data as a trajectory appearing in the KCA and that it holds valid. The same motion is further edited to make it as a double-tap hook motion. The maximum rotation for right upper arm is constrained and the trajectory is well within the KCA of Motion-Sphere as shown in Figure 10(b). It can also be noticed that, since it is a forward hook motion sequence, the trajectory is in the front side of the Motion-Sphere.

V. RESULTS AND DISCUSSION

In this section, we demonstrate the capability of users to employ the proposed system to synthesize realistic

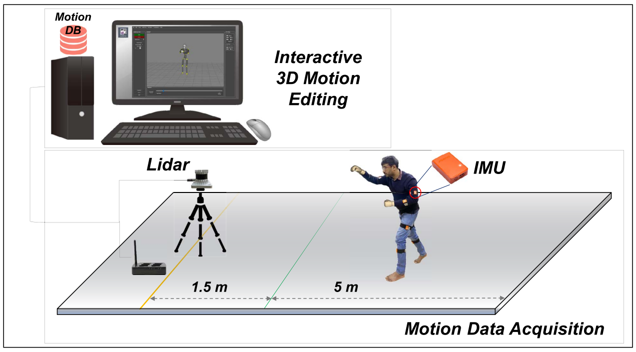


FIGURE 11. Ground truth data acquisition setup: A multi-sensor (LiDAR and IMU) human motion capture environment to record real-time motion data.

human motion data. In addition, we compare the obtained scenario-based edited motion data with those acquired (ground truth) using a real-time motion capture system.

A. EXPERIMENTAL SETUP

We recorded real-time boxing motion sequences using a heterogeneous multi-sensor integrated motion capture system. The motion data acquisition setup consists of a 32-channel light detection and ranging (LiDAR) (Ouster OS-0) system [32] placed at a fixed position and 10 IMU (Xsens MTw Awinda) [30] sensors strapped on to the body segments (pelvis, chest, right upper, and lower arm, left upper and lower arm, right upper and lower leg, and left upper and lower leg) of the subject performing the boxing action. The user performing the boxing action was well within the LiDAR's field of view (1.5 m away from LiDAR system and within a 5 m sensing range), as shown in Figure 11. Furthermore, we estimate the body segment's orientation and joint position using an open-source framework [33]. In this comparative study, we used the estimated orientations and positions from LiDAR measurements and IMU sensors as the ground truth.

B. EDITING SYSTEM USE-CASE EVALUATION

Herein, we illustrate the user's capability to edit the sensed unit motion as an input motion and synthesize three motion variations. Initially, the user needs to select the required motion data (in this experiment, a punch boxing motion data was selected) as shown in Figure 12 (a) from the sensed unit motion database. Next, the user can reconstruct and visualize chosen motion data through a hierarchical humanoid model using our interactive motion editing application system. Subsequently, the user can employ 3D props offered by our application (balls and bars) to compose the user-defined scenarios. In this study, a target hitting scenario was used to synthesize the punch boxing motion variations, namely lower straight, upper straight, and straight mid punch actions (as depicted in Figure 12(b), (c) and (d)). Users can fix the target contact point (green, blue, and purple ball) position and edit the right arm segment's forward-backward motion and lateral swing to make the end effector (right hand) reach the target contact point. Furthermore, the user can adjust the orientations of all other body segments' and pelvis joint position accordingly.

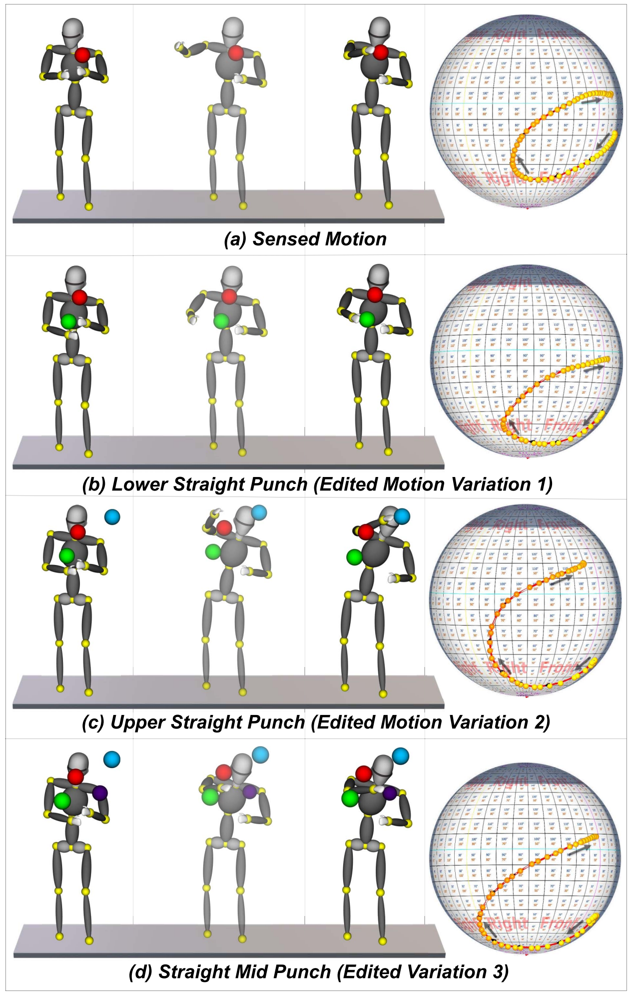


FIGURE 12. Motion editing system use case evaluation: (a) Sensed boxing motion sequence along with the right upper arm segment motion visualization on Motion-Sphere. (b), (c) and (d) represent the three motion variations synthesized using our editing system and their visual validation using Motion-Sphere.

TABLE 1. User interaction attributes to synthesize three variations of the input sensed unit motion.

Motion Variation	Frames	Iterations	Speed	Time
Lower straight punch	190	6	fast	4 *min*
Upper straight punch	280	9	medium	6 *min*
Straight mid punch	510	11	slow	9 *min*

Table 1 shows the user interaction attributes required to compose new variations of sensed input motion. The users can control the editing parameters and complete the editing process in a reasonable time. The time required to compose the poses as well as the number of iterations decrease as the user gets acquainted with our system. The user can ensure the kinematic correctness of the edited motion using the Motion-Sphere's KCA feature.

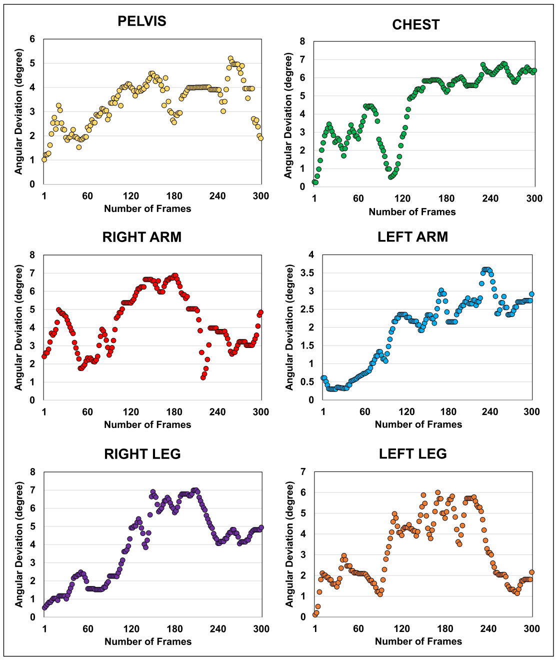


FIGURE 13. Body segments orientation accuracy comparison for first scenario: angular deviation for the pelvis and chest (first row), right and left arm (second row), right and left leg (third row) segments.

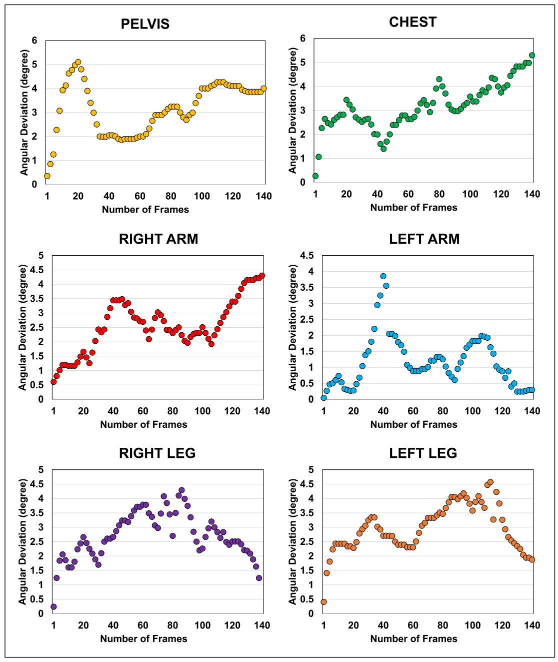


FIGURE 14. Body segments orientation accuracy comparison for second scenario: angular deviation for the pelvis and chest (first row), right and left arm (second row), right and left leg (third row) segments.

C. COMPARATIVE STUDY

The proposed scenario-based sensed human motion editing system is evaluated in terms of orientation and position accuracy. First, three boxing motion scenarios (as in Figure 6(a), (b) and (c)) performed by the subject are captured using the experimental setup shown in Figure 11. Then, we utilized the sensed unit motion data from the motion database to synthesize a set of motion sequences similar to that of real-time motion sequences performed by the subject. Next, we compared the orientations and position of the respective body segments obtained from the sensor setup with those extracted using our editing system.

1) ORIENTATION ACCURACY COMPARISON

We evaluated the orientation estimation accuracy of the proposed editing system against the ground truth data to analyze the similarity between the synthesized motion data and the real-time motion capture system data. We determined the angular difference between the two rotations represented by the unit quaternions using Equation 3 to observe the orientation difference.

$$A_d(q_1, q_2) = 2arccos(q_1^T q_2) \tag{3}$$

where q_1 is the sensor acquired rotation and q_2 is the computed rotation using our system for respective bone segments.

Figure 13, Figure 14, and Figure 15 shows the pelvis, chest, right arm, left arm, right leg, and left leg segment angular deviations between our system in comparison with ground truth, for the three scenarios respectively.

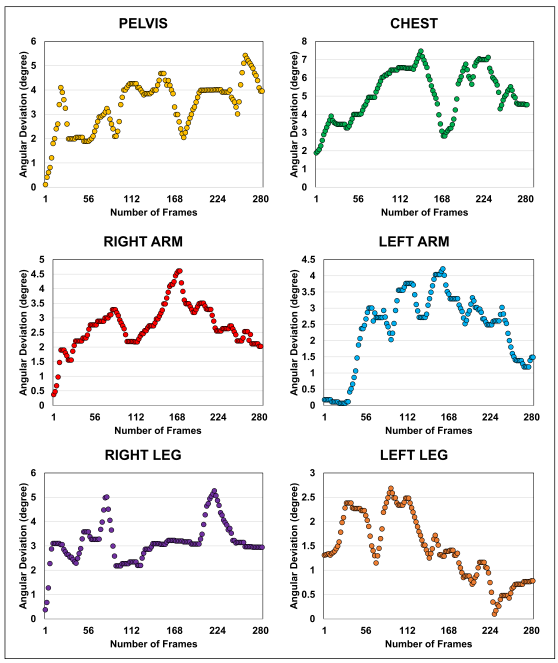


FIGURE 15. Body segments orientation accuracy comparison for third scenario: angular deviation for the pelvis and chest (first row), right and left arm (second row), right and left leg (third row) segments.

Table 2 shows the average angular difference between ground truth data and our system for respective body segments. The example scenarios considered for comparison exhibits a maximum of 3.35° (scenario 2 and 3) for pelvis segment, 5.10° for chest (scenario 3), 4.25° for right arm

TABLE 2. Body segments angular deviation between ground truth and our proposed system.

Body Segment	Angular Deviation in degrees Sensed (Ground Truth) *vs.* Edited (Our System)		
	Scenario 1	Scenario 2	Scenario 3
Pelvis	3.35	3.21	3.35
Chest	4.51	3.17	5.10
Right Arm	4.25	2.53	2.72
Left Arm	1.97	1.18	2.40
Right Leg	4.02	2.74	3.14
Left Leg	3.18	3.05	1.44

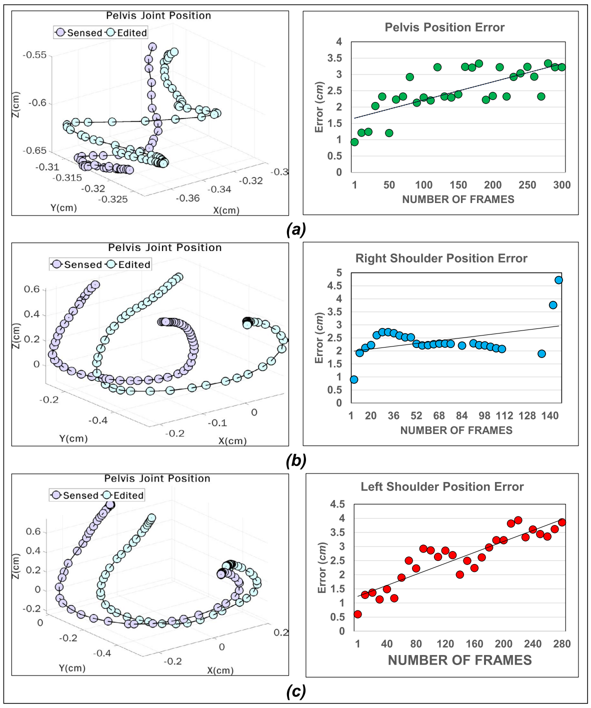


FIGURE 16. Joint position difference comparison: (a) pelvis joint position and its error plot (scenario 1), (b) right shoulder joint position and its error plot (scenario 2), and (c) left shoulder joint position and its error plot(scenario 3).

(scenario 1), 2.40° for left arm (scenario 3), 4.02° for right leg (scenario 1) and 3.18° of angular deviation for left leg (scenario 1) segment. The similarity between the synthesized and sensed motion increases as the user becomes acquainted with the proposed system's motion editing module.

2) POSITION ACCURACY COMPARISON

A vector-based joint position estimation framework [33] was used to evaluate the proposed system's position estimation accuracy. The fullbody skeleton joint positions tracked from the LiDAR's point cloud were fused with those estimated using the IMU sensors. The resultant joint position data were used as the ground truth to validate the position accuracy obtained from the proposed editing system. Figure 16(a), (b) and (c) compares the joint positions (pelvis joint - scenario 1), (right shoulder joint - scenario 2) and (left shoulder joint - scenario 3) in a 3D space and as well as the corresponding errors obtained from our system's data and the ground truth data. The slope depicts the sensor drift error trend.

TABLE 3. Comparison of the joint positions estimated using our system with those obtained from the ground truth.

Body Joint	Mean Position Error in *cm* Sensed (Ground Truth) *vs.* Edited (Our System)		
	Scenario 1	Scenario 2	Scenario 3
Pelvis	2.58	1.65	2.95
Chest	2.40	1.60	2.85
Right Arm	2.65	1.40	2.10
Left Arm	2.05	1.37	2.75
Right Leg	3.25	1.95	2.50
Left Leg	3.58	1.68	2.88

Table 3 presents the mean position error related to the positions estimated using our system, against the real-time captured data.

The example scenarios used for position accuracy comparison exhibits a maximum of 2.95 *cm* error for pelvis (scenario 3), 2.85 *cm* for chest (scenario 3), 2.65 *cm* for right arm (scenario 1), 2.75 *cm* for left arm (scenario 3), 3.25 *cm* for right leg (scenario 1) and 3.58 *cm* error for left leg (scenario 1). The result exhibits a close comparison in accuracy for estimating the body segment orientation and position.

D. DISCUSSION

Human motion analysis and human activity recognition require numerous human motion data sources. Notably, several variations of the same motion are significant in deep learning-based recognition systems. Despite the availability of real-time motion capture systems and human motion databases, extensive efforts are required from the user for capturing the motion sequences every time. Moreover, utilizing the existing motion data and modifying them according to different user requirements is a major challenge.

The proposed open-source scenario-based sensed human motion editing and validation system is a modular tool that can be easily used to synthesize numerous human motion variations using the concept of sensed unit motions. Through this comparative study, the orientation and positioning accuracy of the proposed was evaluated against the real-time motion capture system. The results of this study indicate that it is advantageous to capture minimal motion sequences and employ them further to synthesize a whole new set of motion sequences and numerous subtle variations. Our system also enables the user to validate the kinematic correctness and plausibility of the synthesized human motion visually using the Motion-Sphere. Although the proposed editing system allows the user to manipulate both the orientation and position, it is limited to visually validating the body segment orientation. At present, our system cannot be used to validate the joint position manipulations.

VI. CONCLUSION

With significant advancements in various fields that require human motion analysis and human activity recognition, the demands for human motion data has increased manifold, and thus, it is imperative to synthesize realistic human motion data to cater to all such demands. However, the existing open-source human motion databases exhibit several limitations and are not flexible enough for straightforward usage. Additionally, recording human motion data using a motion capture system in a controlled environment is challenging and tedious. One possible solution is to reuse the existing motion data effortlessly to generate numerous variations of the motion sequences or an entirely new set of motion data.

This paper introduces a comprehensive open-source tool that can be effectively used to author and edit human motion data for synthesizing numerous variations of the existing motion data and an entirely new set of motion sequences. In addition, this tool serves as a basis to validate the kinematic correctness and plausibility of the edited motion data using Motion-Sphere. The concept of sensed unit motion collection, introduced in this paper demonstrates how well this relatively small collection of existing motion datasets can be used to synthesize new motion sequences based on different scenarios. A user-defined scenario serves as a motion plan, and one can select the unit motion segments, do further editing accordingly, and blend them to compose new motion sequences. The end effectors-based analytical kinematic and constraints solver, employed in this study, enables the user to edit the motion data interactively without prior knowledge in a user-friendly environment. Furthermore, the Motion-Sphere serves as a novel and intuitive method to validate the kinematic correctness of the edited motion and thus ensures the naturalness of the synthesized motion data. Finally, we compared the newly synthesized motion data with the IMU motion capture data as the ground truth data to verify the realism of the resultant motion. In the future, we will consider open-source databases and employ the proposed editing and validation framework to synthesize numerous variations of the existing motion data and readily use them in deep learning-based human activity recognition and human motion analysis.

REFERENCES

[1] J. Sedmidubsky, P. Elias, P. Budikova, and P. Zezula, "Content-based management of human motion data: Survey and challenges," *IEEE Access*, vol. 9, pp. 64241–64255, 2021.

[2] C. Mandery, O. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "Unifying representations and large-scale whole-body motion databases for studying human motion," *IEEE Trans. Robot.*, vol. 32, no. 4, pp. 796–809, Aug. 2016.

[3] H. Liu, X. Wei, J. Chai, I. Ha, and T. Rhee, "Realtime human motion control with a small number of inertial sensors," in *Proc. Symp. Interact. 3D Graph. Games (I3D)*, 2011, pp. 133–140.

[4] B. R. Hindle, J. W. L. Keogh, and A. V. Lorimer, "Inertial-based human motion capture: A technical summary of current processing methodologies for spatiotemporal and kinematic measures," *Appl. Bionics Biomech.*, vol. 2021, pp. 1–14, Mar. 2021.

[5] T.-Y. Mou, "Keyframe or motion capture? Reflections on education of character animation," *EURASIA J. Math., Sci. Technol. Educ.*, vol. 14, no. 12, Oct. 2018, Art. no. em1649.

[6] R. Gross and J. Shi, "The CMU motion of body (MoBo) database," Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-01-18, Jun. 2001.

[7] C. Mandery, O. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "The KIT whole-body human motion database," in *Proc. Int. Conf. Adv. Robot. (ICAR)*, Jul. 2015, pp. 329–336.

[8] G. Guerra-Filho and A. Biswas, "The human motion database: A cognitive and parametric sampling of human motion," *Image Vis. Comput.*, vol. 30, no. 3, pp. 251–261, Mar. 2012.

[9] M. Narasimhappa, A. D. Mahindrakar, V. C. Guizilini, M. H. Terra, and S. L. Sabat, "MEMS-based IMU drift minimization: Sage husa adaptive robust Kalman filtering," *IEEE Sensors J.*, vol. 20, no. 1, pp. 250–260, Jan. 2020.

[10] J. Wojtusch and O. von Stryk, "Humod–A versatile and open database for the investigation, modeling and simulation of human motion dynamics on actuation level," in *Proc. IEEE-RAS 15th Int. Conf. Humanoid Robots (Humanoids)*, Nov. 2015, pp. 74–79.

[11] A. Balasubramanyam, A. K. Patil, B. Chakravarthi, J. Y. Ryu, and Y. H. Chai, "Motion-sphere: Visual representation of the subtle motion of human joints," *Appl. Sci.*, vol. 10, no. 18, p. 6462, Sep. 2020.

[12] X. Wang, Q. Chen, and W. Wang, "3D human motion editing and synthesis: A survey," *Comput. Math. Methods Med.*, vol. 2014, Jun. 2014, Art. no. 104535.

[13] K. McHenry and P. Bajcsy, "An overview of 3D data content, file formats and viewers," Nat. Center Supercomput. Appl., Urbana, IL, USA, Tech. Rep. 1205, 2008, p. 22.

[14] Dibya Chakravorty. 2019. *The Most common 3D File Formats*. Accessed: Jun. 12, 2021. [Online]. Available: https://all3dp.com/3d-file-format-3d-files-3d-printer-3d-cad-vrml-stl-obj

[15] Q. Riaz, B. Krüger, and A. Weber, "A relational database for human motion data," in *Proc. Int. Conf. Comput. Sci. Appl.* Cham, Switzerland: Springer, 2015, pp. 234–249.

[16] M. Meredith and S. Maddock, "Motion capture file formats explained," Dept. Comput. Sci., Univ. Sheffield, Sheffield, U.K., Tech. Rep. 211, 2001, pp. 241–244.

[17] S. Ghorbani, K. Mahdaviani, A. Thaler, K. Kording, D. J. Cook, G. Blohm, and N. F. Troje, "MoVi: A large multi-purpose human motion and video dataset," *PLoS ONE*, vol. 16, no. 6, Jun. 2021, Art. no. e0253157.

[18] M. Gleicher, "Animation from observation: Motion capture and motion editing," *ACM SIGGRAPH Comput. Graph.*, vol. 33, no. 4, no. 1999, pp. 51–54.

[19] A. Aristidou, J. Lasenby, Y. Chrysanthou, and A. Shamir, "Inverse kinematics techniques in computer graphics: A survey," *Comput. Graph. Forum*, vol. 37, no. 6, pp. 35–58, 2018.

[20] M. Gleicher, "Comparing constraint-based motion editing methods," *Graph. Models*, vol. 63, no. 2, pp. 107–134, Mar. 2001.

[21] M. Gleicher, "Motion editing with spacetime constraints," in *Proc. Symp. Interact. 3D Graph. (SI3D)*, 1997, p. 139.

[22] J. Lee and S. Y. Shin, "A hierarchical approach to interactive motion editing for human-like figures," in *Proc. 26th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1999, pp. 39–48.

[23] S. Tak, O.-Y. Song, and H.-S. Ko, "Spacetime sweeping: An interactive dynamic constraints solver," in *Proc. Comput. Animation (CA)*, 2002, pp. 261–270.

[24] H. J. Shin and J. Lee, "Motion synthesis and editing in low-dimensional spaces," *Comput. Animation Virtual Worlds*, vol. 17, nos. 3–4, pp. 219–227, 2006.

[25] CMU Graphics Lab Motion Capture Database. *The Data Used in This Project Was Obtained From Mocap.CS.CMU.EDU.The Database Was Created With Funding From NSF EIA-0196217*. Accessed: Jan. 28, 2022. [Online]. Available: http://mocap.cs.cmu.edu/

[26] (2010). *The CMU Multi-Modal Activity Database (CMU-MMAC)*. [Online]. Available: http://kitchen.cs.cmu.edu/

[27] (2015). *The KIT Whole-Body Human Motion Database*. [Online]. Available: https://motion-database.humanoids.kit.edu/list/motions/

[28] B. Chakravarthi, A. K. Patil, A. Balasubramanyam, J. Y. Ryu, and Y. H. Chai, "Sensed unit motion based authoring for the precise human movements," in *Proc. Korean Soc. Mech. Eng. Spring/Autumn Conf.*, 2020, p. 1242.

[29] T. He and A. Kaufman, "Collision detection for volumetric objects," in *Proc. Vis.*, Oct. 1997, pp. 27–34.

[30] M. Paulich, M. Schepers, N. Rudigkeit, and G. Bellusci, "Xsens MTw Awinda: Miniature wireless inertial-magnetic motion tracker for highly accurate 3D kinematic applications," Xsens, Enschede, The Netherlands, Tech. Rep., 2018, pp. 1–9.

[31] A. Balasubramanyam, A. K. Patil, B. Chakravarthi, J. Ryu, and Y. H. Chai, "Kinematically admissible editing of the measured sensor motion data for virtual reconstruction of plausible human movements," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Melbourne, VIC, Australia, Oct. 2021, pp. 283–288.

[32] Ouster. *OS0 Hardware User Manual*. Accessed: Dec. 1, 2020. [Online]. Available: https://data.ouster.io/downloads/hardware-user-manual/hardware-user-manual-revd-os0.pdf

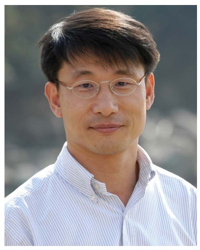
[33] A. K. Patil, A. Balasubramanyam, J. Y. Ryu, B. Chakravarthi, and Y. H. Chai, "An open-source platform for human pose estimation and tracking using a heterogeneous multi-sensor system," *Sensors*, vol. 21, no. 7, p. 2340, Mar. 2021.

BHARATESH CHAKRAVARTHI received the B.E. degree in information science and the M.Tech. degree in computer networks engineering from Visvesvaraya Technological University, Karnataka, India, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree in computer graphics and virtual reality with the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul, South Korea. His research interests include human motion capture systems, human motion visualization, sensor, computer graphics, and virtual reality.

ASHOK KUMAR PATIL received the M.C.A. degree from Visvesvaraya Technological University, Belagavi, India, in 2007, and the Ph.D. degree in CG/VR from the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul, South Korea. He is currently a Postdoctoral Fellow with the Virtual Environment Laboratory, Chung-Ang University. His research interests include computer graphics, virtual reality, interactive systems, HCI, and automation in 3D reconstruction.

JAE YEONG RYU received the master's degree in computer graphics and virtual reality from the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul, South Korea, in 2021, where he is currently pursuing the Ph.D. degree with the Virtual Environments Laboratory. His research interests include virtual reality and motion recognition.

ADITHYA BALASUBRAMANYAM received the Ph.D. degree in computer graphics and virtual reality from Chung-Ang University, Seoul, South Korea. He is currently a Faculty Member with the Department of Computer Science and Engineering, PES University, Bengaluru, India, and a research member with Virtual Environments Lab, Chung-Ang University. His research interests include human motion tracking, virtual reality, and robotics.



YOUNG HO CHAI received the M.S. degree in mechanical engineering from SUNY Buffalo and the Ph.D. degree in mechanical engineering from Iowa State University, in 1997. From 2006 to 2007, he was with the Louisiana Immersive Technology Enterprise (LITE), University of Louisiana at Lafayette, Lafayette, LA, USA. He is currently a Professor at the Graduate School of Advanced Imaging Science, Multimedia, and Film, Chung-Ang University, Seoul, South Korea. He also heads and leads the Virtual Environments Laboratory, Chung-Ang University. His research interests include spatial sketching, HCI, HAR, and motion recognition.