Published in final edited form as:
IEEE Trans Biomed Eng. 2022 February ; 69(2): 678–688. doi:10.1109/TBME.2021.3103201.

An open-source and wearable system for measuring 3D human motion in real-time

Patrick Slade[*]
Department of Mechanical Engineering, Stanford University, Stanford, CA 94305, USA

Ayman Habib, **Jennifer L. Hicks**
Department of Bionengineering, Stanford University.

Scott L. Delp
Departments of Bioengineering and Mechanical Engineering, Stanford University.

Abstract

Objective: Analyzing human motion is essential for diagnosing movement disorders and guiding rehabilitation for conditions like osteoarthritis, stroke, and Parkinson´s disease. Optical motion capture systems are the standard for estimating kinematics, but the equipment is expensive and requires a predefined space. While wearable sensor systems can estimate kinematics in any environment, existing systems are generally less accurate than optical motion capture. Many wearable sensor systems require a computer in close proximity and use proprietary software, limiting experimental reproducibility.

Methods: Here, we present OpenSenseRT, an open-source and wearable system that estimates upper and lower extremity kinematics in real time by using inertial measurement units and a portable microcontroller.

Results: We compared the OpenSenseRT system to optical motion capture and found an average RMSE of 4.4 degrees across 5 lower-limb joint angles during three minutes of walking and an average RMSE of 5.6 degrees across 8 upper extremity joint angles during a Fugl-Meyer task. The open-source software and hardware are scalable, tracking 1 to 14 body segments, with one sensor per segment. A musculoskeletal model and inverse kinematics solver estimate Kinematics in real-time. The computation frequency depends on the number of tracked segments, but is sufficient for real-time measurement for many tasks of interest; for example, the system can track 7 segments at 30 Hz in real-time. The system uses off-the-shelf parts costing approximately $100 USD plus $20 for each tracked segment.

Significance: The OpenSenseRT system is validated against optical motion capture, low-cost, and simple to replicate, enabling movement analysis in clinics, homes, and free-living settings.

Index Terms—

Biomechanics; Inertial Measurement Unit; Kinematics; Wearable Sensors

[*] patslade@stanford.edu .

I. Introduction

Researchers and clinicians measure kinematics during human movement to attain an understanding of diseases and rehabilitation, to improve the design of assistive devices, or to uncover the mechanisms by which athletes achieve peak performance. Estimating kinematics in real-time is important for many real-world applications of movement analysis, such as rapid biomechanics assessment for injury prevention, corrective feedback for at-home rehabilitation, or real-time control of assistive devices to improve mobility.

Motion capture systems use sensors to objectively measure the motion of body segments and estimate joint angles and other kinematic quantities. Motion capture systems typically rely on either stationary sensors such as cameras placed throughout a room or wearable sensors such as inertial measurement units (IMUs) attached to the body. Systems that measure joint kinematics with less than 2 degrees root-mean-square error (RMSE) are considered to be excellent and less than 5 degrees RMSE to be acceptable for many applications [1], although accuracy requirements depend on the application.

Optical motion capture is the current industry standard. Careful calibration can enable these camera systems to track the position of markers with millimeter accuracy. However, capturing human motion adds additional sources of error such as skin motion artifacts and occluded markers [2], which can increase the average RMSEs for joint angles by 1 to 3 degrees [3]. The use of optical motion capture is further limited by the high cost of the equipment and the requirement of a fixed capture volume.

Wearable systems use portable sensors, most commonly IMUs, worn on the body to capture motion in many environments. Modern IMUs measure linear acceleration, angular velocity, and the local magnetic field and are compact, lightweight, and low-cost compared to optical motion capture equipment. Joint kinematics can be computed with a variety of methods including integration [4], constrained orientation [5], statistical models [6], or musculoskeletal simulation [7]. Commercial IMU motion capture systems can estimate joint kinematics with less than 5 degrees RMSE compared to optical motion capture during brief bouts of activities including walking, jumping, and squatting [8]. Most current wearable systems estimate kinematics offline due to the significant number of processing steps and computation requirements. Computing kinematics in real-time is necessary for providing rapid results or diagnoses as well as applications that require feedback to the wearer, such as correcting a rehabilitation exercise [9] or improving an athletic maneuver [10].

Beyond computing kinematics in real-time, wearable systems should meet several additional criteria to be effective across a wide range of real-world scenarios. First, the accuracy and frequency of estimation must be sufficient for the application of interest. Systems that require off-board computation, such as a laptop or desktop, to provide real-time estimates restrict the motion capture volume to the range where the sensors can communicate with the computer. Thus, on-body computation is preferred for use in a wide range of environments. A wearable motion capture system for real-world applications also requires robustness to environmental changes such as the magnetic field, which can skew magnetometer measurements, and temperature, which can affect gyroscope measurements. Design guidelines for wearable medical devices suggest devices should be lightweight, low-cost, and easy to don and doff for extended use [11]. In addition, systems with open-source software, simple assembly procedures, and a straightforward calibration method would reduce barriers to use.

Existing real-time wearable systems span a range of features (Table I). All of these systems require off-board computation, which increases the overall system cost and limits motion capture to the range in which sensor data can be passed to the computer. Further, existing systems either use custom hardware and are not open-source, use commercial hardware and are expensive, or compute kinematics for a fixed set of joints for a specific application. While current commercial systems meet some of the desired specifications for real-time wearable systems such as accurately estimating joint kinematics, high computation frequency, and flexibility in the number of tracked body segments, they are expensive and require the user to stay within approximately 40 feet of a desktop or laptop computer.

To help overcome the current barriers to widespread use of real-time wearable systems, we developed the OpenSenseRT system, a low-cost, open-source, and fully wearable system for computing real-time kinematics. We assessed accuracy by comparing the OpenSenseRT system to optical motion capture for a range of upper and lower extremity movements, seeking to achieve an RMSE of 5 degrees or less for 3D joint angles. To make the system useful in a wide range of applications we used off-the-shelf components and enabled customization to estimate any combination of lower and upper extremity motions (assuming one sensor worn on each tracked segment). We have provided the software, a list of components, instructions for assembly and use, and examples of activities measured with the OpenSenseRT system [20].

II. Methods

A. Open-source system hardware

The system hardware is built from components that are inexpensive and can be assembled without tools or soldering. The components include IMUs, a microcontroller, Qwiic connector cables, a button (to start and stop recording), and a USB-rechargeable battery (Fig. 1). The sizes of the IMU, microcontroller assembly, and battery are 0.2 by 1.6 by 2.5 cm, 3.5 by 6 by 9 cm, and 2.5 by 5 by 10.3 cm, respectively. At the time of publication, parts to build the system cost approximately $100 plus $20 for each tracked body segment. The assembly procedure consists of assembling a case for the microcontroller and fan, attaching this case to a belt to be worn on the pelvis, and connecting the sensors to the microcontroller. A complete bill of materials, suggested vendors, written assembly instructions, and video of the assembly procedure are provided in a user’s guide to simplify replication of the system [20].

The microcontroller and IMUs were selected because they meet computational requirements for real-time estimation of joint kinematics. The microcontroller is a Raspberry Pi 4b+ with 4 gigabytes of RAM and a quad-core microprocessor (Raspberry Pi Foundation). This amount of RAM is required to build the OpenSim software [21] that performs the inverse kinematics. The quad-core microprocessor allows for multiple threads to run in parallel, necessary for the system's software architecture discussed in the next section. The IMUs are ISM330DHCX breakout boards (Adafruit) which contain an accelerometer and gyroscope. These IMUs measure accelerations up to 16 times gravity and angular velocities up to 4000 degrees per second, sufficient to capture dynamic activities like running. The analog to digital resolution of the IMU is 16 bits. The IMU sampling rate is variable with a maximum rate of 6.7 kHz. An IMU was measured in a static position for 60 seconds to determine the standard deviations of the accelerometer measurements, which were 0.0155, 0.0244, and 0.254 meters per second squared, and the gyroscope which were 0.0742, 0.0583, and 0.0369 degrees per second for the x, y, and z directions, respectively. The IMU breakout board includes connectors to simplify wiring the IMUs to the microcontroller with pre-made cables. We recommend gluing the pre-made cables into connectors on the IMU and microcontroller to improve the strength of the connections.

The hardware system specifications are designed to estimate joint kinematics for extended periods in free-living settings. The total weight of the system is 408 grams, with approximately half due to the rechargeable battery. The rechargeable battery allows continuous recording for approximately 6.5 hours per charge and can be recharged in about 2 hours or replaced for continual use. The SD card in the microcontroller is able to store approximately 100 hours of recordings at a 30 Hz computation rate. A button and blinking LED provide control and feedback to the user to start and stop recordings.

B. Open-source software architecture

The open-source software requires user inputs to define which body segments are tracked with IMUs and uses a two phase procedure to estimate kinematics: initial calibration and iterative computation of joint kinematics (Fig. 2). The software builds on the OpenSense tools for IMU-based kinematics [22] that are part of OpenSim version 4.2 [21]. The microcontroller's SD card contains the operating system for the OpenSenseRT system. The user-defined parameters are input to the system by editing a text file on the SD card. Step-by-step instructions to edit this text file and additional instructions to customize the software are provided in the user's guide [20].

The software performs a calibration of the gyroscopes in the IMUs during their first use. Gyroscope measurements are taken for 10 seconds and averaged to remove any offset, which is a standard IMU processing technique to prevent drift due to sensor bias. The gyroscope offset fluctuated less than 0.00571 degrees per second when measured for several minutes, thus only 10 seconds of data was required to accurately estimate the gyroscope offset. IMUs with similar gyroscope noise levels are expected to perform similarly. These gyroscope bias values are stored and recalled for future recordings with the same set of sensors. The gyroscope bias was measured 3 days in a row to understand the variation over time. The largest difference between the bias measurements across these days was 0.0172, 0.0247, and 0.0178 degrees per second for the x, y, and z directions. These differences were below the 0.1 degrees per second standard deviation in gyroscope measurement, indicating the gyroscope bias values do not degrade measurement accuracy over a few days. Periodic recalibration of the gyroscopes is important to minimize drift in the sensors. The system will automatically calibrate the gyroscope measurements when the user edits the text file to tell the system to recalibrate, following the instructions in the user's guide [20].

Before each recording session, a system calibration must be performed to determine the initial orientation of all the IMUs; we assume a standing pose and a set of known IMU orientations with respect to each body segment. The subject stands in a neutral position with their arms at their sides, with all IMUs aligned as shown in Fig. 1. The initial orientation of the IMUs is computed from this stationary standing data by using a Mahony Filter [23]. Since OpenSenseRT does not use magnetometer measurements, the initial heading angle of the IMUs with respect to the anterior-posterior axis of the musculoskeletal model is randomly rotated to point in different directions. First, we compute the average heading angle across all IMUs which we will use as our target heading angle. Then we align all IMUs to the same initial heading angle by rotating each IMU orientation by the difference between its heading angle and the target heading angle. Thus, all the IMUs are initialized with the same target heading angle. The IMUs now match the initial generalized coordinates of the musculoskeletal model used to estimate kinematics.

Once calibrated, the system uses the inverse kinematics solver of OpenSim to compute new joint kinematics at each timestep based on the IMU orientations. An IMU on the pelvis is required and acts as the base segment against which we compute the relative orientation of other sensors. The microcontroller uses two separate threads to compute joint kinematics at each step. The first thread samples each IMU and converts the raw accelerometer and gyroscope readings into a quaternion containing the orientation by using a Mahony Filter [23]. Pilot walking data was used to compare the joint kinematic estimates using a Mahony Filter to optical motion capture estimates in order to select the appropriate parameters for the Mahony Filter. Although a range of proportional and integral gains were evaluated, all sets of parameters resulted in errors within 0.1 degrees RMSE of each other. Thus, the proportional gain was left to the recommended default value of 1, and an integral gain of 0.3 was chosen empirically to allow the initial IMU orientations to reach a steady-state value within 3 seconds. Further tuning of these parameters may improve IMU orientation estimates for specific applications [24], [25]. Magnetometer measurements were not used to compute the quaternions, because they are sensitive to changes, for example, when near electric motors or large metal objects. The first thread operates approximately 12% of the time, while the second thread runs continuously. Thus, the computational frequency is limited by the speed of the inverse kinematics solver.

The second thread uses these orientations and the musculoskeletal model as inputs to the inverse kinematics solver included in OpenSim version 4.2 [21], [22] to compute the generalized coordinates of the model (i.e., joint angles and joint angular velocities). The inverse kinematics solver uses the generalized coordinates of the musculoskeletal model at the previous time step as an initial guess. The new set of generalized coordinates are computed by taking the error relative to the measured IMU orientations from the current step, converting to an axis-angle representation, and then minimizing this angle-axis difference. The axis-angle representation defines orientation in three dimensions with a unit vector, or axis, and a single angle about this axis. These steps are repeated to estimate the kinematics as measurement moves forward in time. Using two threads reduces the delay in computing kinematics, allowing for higher computation frequencies. For the experimental comparison described below, the generic musculoskeletal model did not need to be scaled to subject anthropometry, because we only examined joint kinematics that depended on relative orientations of the segments.

Estimating kinematics in real-time required several modifications to the musculoskeletal model and inverse kinematics settings. We modified a previously published full-body musculoskeletal model [26] by removing muscles and locking unused joints, including the subtalar and metatarsophalangeal joints. The inverse kinematics solver used a weight on the joint constraints of 10 and a solver tolerance of 0.001. These values were selected through trial and error in pilot testing of the device to achieve both accurate and fast solutions.

The OpenSenseRT system can estimate kinematics online or offline. The online mode uses one or more of the microcontroller cores to compute kinematics. The offline mode only samples the IMUs, performing the inverse kinematics computation later. The offline mode provides much faster sampling frequencies by removing the need to compute kinematics in real-time. Higher frequency estimates of joint kinematics improve the accuracy when estimating dynamic motions such as running and may be preferred for applications without real-time constraints.

C. Experimental comparison to optical motion capture

We compared estimates of joint kinematics from the OpenSenseRT system to estimates based on optical motion capture (Optitrack) for a range of conditions including: walking, running, a Fugl-Meyer upper-limb task [27], and a task measuring the range of motion of the trunk. The motion capture was recorded at 100 Hz. Lower-limb experiments used sixteen markers, including markers bilaterally on the anterior and posterior superior iliac spines, as well as markers on the right leg with a three-marker plate on the thigh, a four-marker plate on the shank, and single markers on the femoral epicondyles, malleoli, calcanei, and the 2nd and 5th metatarsal heads. Upper-limb experiments used 14 markers bilaterally on the anterior and posterior superior iliac spines, as well as a marker on the C7 vertebrae and markers on the right side of the body for the clavicle, anterior and posterior shoulder, lateral shoulder, lateral and medial epicondyles of the humerus, radial and ulnar styloid processes, and the body of the 5th metacarpal. Healthy young adults ($n = 5$, 3 men and 2 women; age = 25.0 ± 1.8 yr; body mass = 63.0 ± 10.3 kg; height = 1.70 ± 0.07 m) were recruited. All subjects were volunteers and provided written informed consent before completing the protocol IRB-17282 approved by the Stanford University Institutional Review Board. Only one subject completed the running condition, but all subjects completed all other conditions. The walking condition lasted three minutes on a treadmill at 1.25 m/s. The running condition lasted 1 minute on a treadmill at 4.0 m/s. The Fugl-Meyer upper-limb and trunk range of motion conditions were completed ten times to create a cyclic upper-limb task for analyzing an average activity cycle in a similar manner to a gait cycle [28]. The Fugl-Meyer upper-limb task simulated cutting by having the subject pick up a knife from a table, perform a cutting motion, and place the knife back on the table [29]. The range of motion of the trunk task had subjects reach their comfortable range of motion for trunk flexion, then rotation, then bending [30].

The optical motion capture data was input to the OpenSim software [21], [22] to estimate the ground truth kinematics. The optical motion capture data was processed following best practices to ensure proper model scaling, an accurate initial pose, and precise tracking of the marker positions using OpenSim's inverse kinematics solver [31]. The joint kinematics computed with the OpenSenseRT system utilized the previously described open-source hardware and software. The similarity between joint kinematics computed with optical motion capture and OpenSenseRT was quantified with the Pearson correlation coefficient [32].

While our validation focused on joint kinematics, the OpenSenseRT system can compute any quantity that is derived from the kinematic state of the model. For demonstration purposes, we computed foot progression angle. We first computed virtual marker positions on the heel and second metatarsal head, using OpenSim's analyze tool offline. Then the foot progression angles were computed from these virtual marker positions with a previously defined approach [33]. The OpenSenseRT system could perform this and other kinematic analyses in real time but would require custom development for a given application.

III. Results

A. Accuracy of estimated joint kinematics

During walking, the OpenSenseRT system estimated lower-limb joint kinematics with an average RMSE of 4.4 degrees compared to kinematics computed from optical motion capture data. Joint kinematic curves from the OpenSenseRT system were compared to optical motion capture for both walking (Fig. 3A) and running (Fig. 3B). The Pearson correlation coefficients between the OpenSenseRT system and optical motion capture were above 0.95 for all joint angles other than hip rotation during both walking and running (Table II). The joint kinematic curves from OpenSenseRT had a larger standard deviation than optical motion capture for the running trials, indicating the system was more susceptible to fluctuations during faster motion or higher impact activities. The absolute RMSE averaged across all joints drifted from 3.5 to 6 degrees after 3 minutes, although most angles stayed below a RMSE of 5 degrees for the first minute of data capture (Fig. 4). Hip rotation had the largest RMSE among the joint angles measured but also has larger errors relative to other joints when measured with optical motion capture [3].

The OpenSenseRT system estimated upper-limb joint kinematics with an average RMSE of 5.6 degrees compared to kinematics computed from optical motion capture. The individual upper-limb joint angles had average RMSEs between 2.5 and 10 degrees (Fig. 5). Average joint kinematic curves from the OpenSenseRT system were similar to values from optical motion capture for the upper-limb task (Fig. 6A). The joint kinematics during the upper-limb task had larger standard deviations than other conditions for both OpenSenseRT and optical motion capture, because the activity cycles occurred over a longer time period and the movement was more variable across repetitions and subjects. The trunk joint kinematics estimates during the trunk range of motion task for the OpenSenseRT system and optical motion capture had similar means and standard deviations (Fig. 6B). The Pearson correlation coefficients between the OpenSenseRT system and optical motion capture were above 0.87 for all joint angles other than shoulder rotation (Table III).

Foot progression angle was computed to evaluate an example of a clinically relevant metric that is derived from the generalized coordinates of the model. The estimated foot progression angle for a single individual showed similar trends to the values computed based on optical motion capture data as the angle was varied during walking (Fig. 7).

B. Computational limitations

We evaluated the real-time computational limits of the OpenSenseRT system to understand the relationship between the number of sensors used to track body segments and the computation rate of the inverse kinematics solver. We evaluated the fastest computation rate for the system when using 1 to 14 sensors. We found the maximum computation rate when using a single core of the microcontroller had a line of best fit of $69 - 4.8 * N$, with N denoting the number of IMU sensors used to compute the kinematics (Fig. 8A). This recommended maximum computation rate should be a guideline when customizing the number of IMU sensors for new applications. We expect a similar level of performance for other IMUs with sampling rates faster than the desired computation rate of the system. When computing kinematics using a single core, the microcontroller had a steady-state CPU temperature safely below the soft and hard temperature limits set by the operating system (Fig. 8B). Thus, the device can safely operate in this mode indefinitely without encountering computation limits due to heat. These computation experiments were performed indoors at a room temperature of approximately 22 degrees Celsius. We anticipate the single core mode of operation results reported here could be replicated in ambient temperatures up to 37 degrees Celsius without performance degradation.

A second real-time mode of operation utilized all cores on the microcontroller to increase the maximum computation at the expense of increased CPU temperature. The maximum computation rate in this mode had a line of best fit of $93 - 5.6 * N$, where N denotes the number of IMU sensors. The CPU temperature reached the soft limit after 92 seconds and the hard limit after 10 minutes. The soft computation limit reduced the microcontroller performance and computation rate, preventing computation of joint kinematics in real time. Thus, using all cores may provide a faster real-time computation rate than a single core, but cannot be sustained without overheating the microcontroller.

An offline mode was tested by taking measurements from the IMUs in real-time and then computing the kinematics once the recording was finished. The computation in real-time was only limited by how fast the sensors could be sampled, allowing much faster offline computation rates of $150 – 7.5 * N$, where N is the number of IMU sensors. This mode offered sampling frequencies similar to other commercial systems [7] capable of capturing dynamic activities such as running.

IV. Discussion

The OpenSenseRT system is the first open-source, low-cost system to estimate kinematics for a range of different movements in real time. We computed kinematics for 14 different body segments, including upper- and lower-extremity joint kinematics, with an average RMSE of 5 degrees compared to optical motion capture. The OpenSenseRT system provides flexibility, with high-frequency sampling, real-time or offline kinematics computation, and access to the modeling and analysis tools available in OpenSim, which we demonstrate by computing foot progression angle. The system provides technology to estimate joint kinematics in free-living conditions, which will help translate clinical tools into the real-world and enable large-scale and long-term biomechanics studies.

A. Evaluating the accuracy of IMU motion capture

Joint kinematics estimated by the OpenSenseRT system had an average error of 5 degrees RMSE across lower and upper-extremity joints compared to optical motion capture, an acceptable error for use in many clinical and research applications [34]. Pearson correlation coefficients were also high (Tables II and III), indicating that the two methods produced similar kinematic profiles. Accuracy requirements will vary and should be validated for a specific clinical application before using the OpenSenseRT system to evaluate clinical outcomes. The error bands of hip and knee flexion during walking had similar magnitudes for both the OpenSenseRT system and optical motion capture, indicating that estimated kinematics were similarly consistent for both approaches. We expect there to be differences between kinematics estimates with OpenSenseRT and optical motion capture because optical motion capture is known to have measurement errors, for example due to skin motion artifacts. The error and Pearson correlation coefficients between optical motion capture and the OpenSenseRT system for lower-limb joint kinematics were within the range of several state-of-the-art commercial systems which compute kinematics offline, including Xsens [13], [35] and other systems [36]. Two offline systems reported lower errors during walking [37], [38]. The upper-limb estimates error and Pearson correlation coefficients were also within the range of all IMU systems that compute kinematics offline [35], [39]–[41] except for one system that reported smaller errors [42]. The higher errors associated with the upper-limb joints may be due to their larger range of motion and increased degrees of freedom compared to the lower limbs.

B. Feasibility for estimating kinematics during real-world use

The OpenSenseRT system has usability and technical specifications that support estimating kinematics in real-world settings. The system is designed for continual monitoring applications: it is lightweight to minimize fatigue, it has a rechargeable battery capable of recording for most of a day, it can store days or weeks of kinematic estimates in memory, and it allows for recordings to be manually started and stopped with the press of a button. The system is also meant to be easily reproducible–the device consists of low-cost and off-the-shelf components that can be assembled in approximately 20 minutes using common tools without soldering or coding [20].

The error in joint kinematic estimates drifted over time as expected, since our sensor fusion approach did not use magnetometer information. Investigation into the sources of the drift, such as the heading or inclination estimates of the IMU [43], as well as the variability among the IMU sensors might help reduce this drift. When using the OpenSenseRT system for extended recordings, we suggest recalibrating the system periodically to reduce the average error. For example, in an all day monitoring experiment it may be possible to automatically re-calibrate the OpenSenseRT system by trying to detect when a quiet standing pose occurred.

C. Extending the system to custom applications

The OpenSenseRT relies on open-source software and hardware for transparency and extension to new applications. The software builds upon OpenSim [21], a software framework with well documented musculoskeletal analysis and simulation tools that can be integrated for future uses, such as computing muscle-tendon lengths and velocities. Prior research has shown that IMUs can be used with musculoskeletal analysis tools to estimate ground reaction forces [44] and joint kinetics [45], [46]. Although the use of a physiological skeletal model can be more computationally intensive than other approaches for computing kinematics, it allows kinematics estimates to be constrained to realistic joint limits and provides access to other biomechanics tools that rely on musculoskeletal modeling [21]. The open-source software also allows for custom additions to compute IMU metrics such as energy expenditure [47] or spatiotemporal metrics such as gait cycle duration (e.g., using the approach of [48]), which are important for clinical analysis of lower-limb motion. The OpenSenseRT system provides hardware flexibility by allowing additional sensors, such as pressure insoles or electromyography sensors, to be connected to the microcontroller and open-source hardware, enabling customized measurements.

D. Limitations

We have benchmarked the OpenSenseRT system’s basic functionality, but additional work is required to validate the system for more challenging applications, such as activities that require measuring both the upper- and lower-limbs or dynamic activities with a large range of motion or high speeds, such as squatting or sprinting. The computation rate is limited by the performance of the microcontroller, which prevents fast computation rates with many sensors, making a task like estimating full-body kinematics during running challenging. The CPU speed may be increased by overclocking the microcontroller or adding additional computation sources, such as a USB accelerator. Using all cores of the microcontroller significantly increased the computation rate, but would require additional cooling to prevent overheating.

The accuracy of the OpenSenseRT system is dependent on the initial calibration which requires firmly and precisely attaching the IMUs to each body segment, which could pose challenges for real-world use. Methods for simplifying the calibration process and generalizing to any initial pose would allow for easier recalibration. In order to accurately capture motion, the IMUs need to be firmly attached to the body as any shift in their placement accumulates error in the joint kinematics over time. Integrating the IMUs into clothes may also improve consistency for monitoring applications over extended periods of time. Accurately estimating the positions of body segments, such as the hand or foot would require making the lengths of the musculoskeletal model to match the subject. The IMUs do not require the musculoskeletal model to be scaled to each subject, but the model can be scaled, if desired.

V. Conclusion

OpenSenseRT is an open-source, low-cost, easy-to-assemble, and customizable system for computing real-time kinematics. This system will enable researchers to translate biomechanics, rehabilitation, and sports performance research that rely on kinematics into real-world solutions. The system's accuracy meets clinical standards for several common use cases in biomechanics and rehabilitation research. We have also documented and openly shared all hardware and software to allow other researchers to use the system in a wide range of new applications that leverage the biomechanics software tools included in OpenSim. For example, wearable and real-time joint kinematics will allow research to test biofeedback for gait retraining or sports performance outside of the lab, and the low cost could enable large-scale and longitudinal collection of free-living biomechanics data. The accessibility of this technology could also reduce the financial and technological barriers to perform state-of-the-art quantitative biomechanical analysis in low-income or developing countries with limited access to laboratory equipment, offering tools to improve musculoskeletal health on a global scale.

Acknowledgment

The authors would like to acknowledge the staff of the Human Performance Lab for supporting the equipment used for the experimental validation of this work.

This work was supported by the National Science Foundation Graduate Research Fellowship Program Grant DGE-1656518; The Mobilize Center, a Biomedical Technology Resource Center, funded by the NIH National Institute of Biomedical Imaging and Bioengineering through Grant P41EB027060; and the Restore Center, which is funded by the Eunice Kennedy Shriver National Institute Of Child Health & Human Development (NICHD) and the National Institute Of Neurological Disorders And Stroke (NINDS) of the National Institutes of Health (NIH) under Grant Number P2CHD101913.

References

[1]. McGinley Jennifer L et al. The reliability of three-dimensional kinematic gait measurements: A systematic review. Gait & Posture, 29(3):360–369, 2009. [PubMed: 19013070]

[2]. Reinschmidt Christoph et al. Effect of skin movement on the analysis of skeletal knee joint motion during running. Journal of Biomechanics, 30(7):729–732, 1997. [PubMed: 9239553]

[3]. Leardini Alberto et al. Human movement analysis using stereophotogrammetry: Part 3. soft tissue artifact assessment and compensation. Gait & Posture, 21(2):212–225, 2005. [PubMed: 15639400]

[4]. Kong Weisheng et al. Development of a real-time IMU-based motion capture system for gait rehabilitation. In 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 2100–2105. IEEE, 2013.

[5]. Zheng Yang et al. Pedalvatar: An IMU-based real-time body motion capture system using foot rooted kinematic model. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4130–4135. IEEE, 2014.

[6]. Del Rosario Michael B et al. Computationally efficient adaptive error-state Kalman filter for attitude estimation. IEEE Sensors Journal, 18(22):9332–9342, 2018.

[7]. Roetenberg Daniel et al. Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors. Xsens Motion Technologies BV, Tech. Rep, 1, 2009.

[8]. Mohammad Al-Amri et al. Inertial measurement units for clinical movement analysis: Reliability and concurrent validity. Sensors, 18(3):719, 2018.

[9]. Zedda A et al. Domomea: a home-based telerehabilitation system for stroke patients. In 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), pages 5773–5776. IEEE, 2020.

[10]. Escarti Amparo and Guzman Jose F. Effects of feedback on self-efficacy, performance, and choice in an athletic task. Journal of Applied Sport Psychology, 11(1):83–96, 1999.

[11]. Mosenia Arsalan et al. Wearable medical sensor-based system design: A survey. IEEE Transactions on Multi-Scale Computing Systems, 3(2):124–138, 2017.

[12]. Zhou Huiyu et al. Inertial measurements of upper limb motion. Medical and Biological Engineering and Computing, 44(6):479–487, 2006. [PubMed: 16937199]

[13]. Schepers Martin et al. Xsens MVN: Consistent tracking of human motion using inertial sensing. Xsens Technol, pages 1–8, 2018.

[14]. Maruyama Tsubasa et al. Constraint-based real-time full-body motion-capture using inertial measurement units. In 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pages 4298–4303. IEEE, 2018.

[15]. Miller Nathan et al. Motion capture from inertial sensing for untethered humanoid teleoperation. In 4th IEEE/RAS International Conference on Humanoid Robots, 2004., volume 2, pages 547–565. IEEE, 2004.

[16]. Andrews Sheldon et al. Real-time physics-based motion capture with sparse sensors. In Proceedings of the 13th European Conference on Visual Media Production (CVMP 2016), pages 1–10, 2016.

[17]. Raghavendra Padmanabha et al. Design and development of a real-time, low-cost IMU based human motion capture system. In Computing and Network Sustainability, pages 155–165. Springer, 2017.

[18]. Zhang Yang et al. Micro-IMU-based motion tracking system for virtual training. In 2015 34th Chinese Control Conference (CCC), pages 7753–7758. IEEE, 2015.

[19]. Sy Luke et al. Estimating lower limb kinematics using a reduced wearable sensor count. arXiv preprint arXiv:1910.00910, 2019.

[20]. Slade Patrick et al. Github link for data repository and replication instructions. https://simtk-confluence.stanford.edu/display/OpenSim/Wearable+and+Real-time+Kinematics+Estimates+with+OpenSense, 2021.

[21]. Seth Ajay et al. Opensim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. PLoS Computational Biology, 14(7):e1006223, 2018. [PubMed: 30048444]

[22]. Borno Mazen Al et al. Opensense: An open-source toolbox for inertial-measurement-unit-based measurement of lower extremity kinematics over long durations. bioRxiv, 2021.

[23]. Euston Mark et al. A complementary filter for attitude estimation of a fixed-wing UAV. In 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 340–345. IEEE, 2008.

[24]. Caruso Marco et al. Orientation estimation through magneto-inertial sensor fusion: A heuristic approach for suboptimal parameters tuning. IEEE Sensors Journal, 21(3):3408–3419, 2020.

[25]. Caruso Marco et al. Analysis of the accuracy of ten algorithms for orientation estimation using inertial and magnetic sensing under optimal conditions: One size does not fit all. Sensors, 21(7):2543, 2021. [PubMed: 33916432]

[26]. Rajagopal Apoorva et al. Full-body musculoskeletal model for muscle-driven simulation of human gait. IEEE Transactions on Biomedical Engineering, 63(10):2068–2079, 2016. [PubMed: 27392337]

[27]. Fugl-Meyer Axel R et al. The post-stroke hemiplegic patient. 1. a method for evaluation of physical performance. Scandinavian Journal of Rehabilitation Medicine, 7(1):13–31, 1975. [PubMed: 1135616]

[28]. Butler Erin E et al. Three-dimensional kinematics of the upper limb during a reach and grasp cycle for children. Gait & Posture, 32(1):72–77, 2010. [PubMed: 20378351]

[29]. Sullivan Katherine J et al. Fugl-Meyer assessment of sensorimotor function after stroke: Standardized training procedure for clinical practice and clinical trials. Stroke, 42(2):427–432, 2011. [PubMed: 21164120]

[30]. Bonato Paolo et al. Muscle fatigue and fatigue-related biomechanical changes during a cyclic lifting task. Spine, 28(16):1810–1820, 2003. [PubMed: 12923468]

[31]. Hicks Jennifer L et al. Is my model good enough? Best practices for verification and validation of musculoskeletal models and simulations of movement. Journal of Biomechanical Engineering, 137(2), 2015.

[32]. Benesty Jacob et al. Pearson correlation coefficient. In Noise reduction in speech processing, pages 1–4. Springer, 2009.

[33]. Rutherford DJ et al. Foot progression angle and the knee adduction moment: A cross-sectional investigation in knee osteoarthritis. Osteoarthritis and Cartilage, 16(8):883–889, 2008. [PubMed: 18182310]

[34]. Del Rosario Michael B et al. Quaternion-based complementary filter for attitude determination of a smartphone. IEEE Sensors Journal, 16(15):6008–6017, 2016.

[35]. Mavor Matthew P et al. Validation of an imu suit for military-based tasks. Sensors, 20(15):4280, 2020.

[36]. Bolink Saan et al. Validity of an inertial measurement unit to assess pelvic orientation angles during gait, sit–stand transfers and step-up transfers: Comparison with an optoelectronic motion capture system. Medical engineering & physics, 38(3):225–231, 2016. [PubMed: 26711470]

[37]. Nüesch Corina et al. Measuring joint kinematics of treadmill walking and running: Comparison between an inertial sensor based system and a camera-based system. Journal of biomechanics, 57:32–38, 2017. [PubMed: 28366438]

[38]. Zhang Jun-Tian et al. Concurrent validation of xsens mvn measurement of lower limb joint angular kinematics. Physiological measurement, 34(8):N63, 2013. [PubMed: 23893094]

[39]. Morrow Melissa MB et al. Validation of inertial measurement units for upper body kinematics. Journal of Applied Biomechanics, 33(3):227–232, 2017. [PubMed: 27918696]

[40]. Filippeschi Alessandro et al. Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion. Sensors, 17(6):1257, 2017.

[41]. Wirth Michael Alexander et al. Comparison of a new inertial sensor based system with an optoelectronic motion capture system for motion analysis of healthy human wrist joints. Sensors, 19(23):5297, 2019.

[42]. Sers Ryan et al. Validity of the perception neuron inertial motion capture system for upper body motion analysis. Measurement, 149:107024, 2020.

[43]. Bergamini Elena et al. Estimating orientation using magnetic and inertial sensors and different sensor fusion approaches: Accuracy assessment in manual and locomotion tasks. Sensors, 14(10):18625–18649, 2014. [PubMed: 25302810]

[44]. Karatsidis Angelos et al. Estimation of ground reaction forces and moments during gait using only inertial motion capture. Sensors, 17(1):75, 2017.

[45]. Konrath Jason M et al. Estimation of the knee adduction moment and joint contact force during daily living activities using inertial motion capture. Sensors, 19(7):1681, 2019.

[46]. Dorschky Eva et al. Estimation of gait kinematics and kinetics from inertial sensor data using optimal control of musculoskeletal models. Journal of biomechanics, 95:109278, 2019. [PubMed: 31472970]

[47]. Slade Patrick et al. Sensing leg movement enhances wearable monitoring of energy expenditure. Nature Communications, 12(1):1–11, 2021.

[48]. Jasiewicz Jan M et al. Gait event detection using linear accelerometers or angular velocity transducers in able-bodied and spinal-cord injured individuals. Gait & Posture, 24(4):502–509, 2006. [PubMed: 16500102]

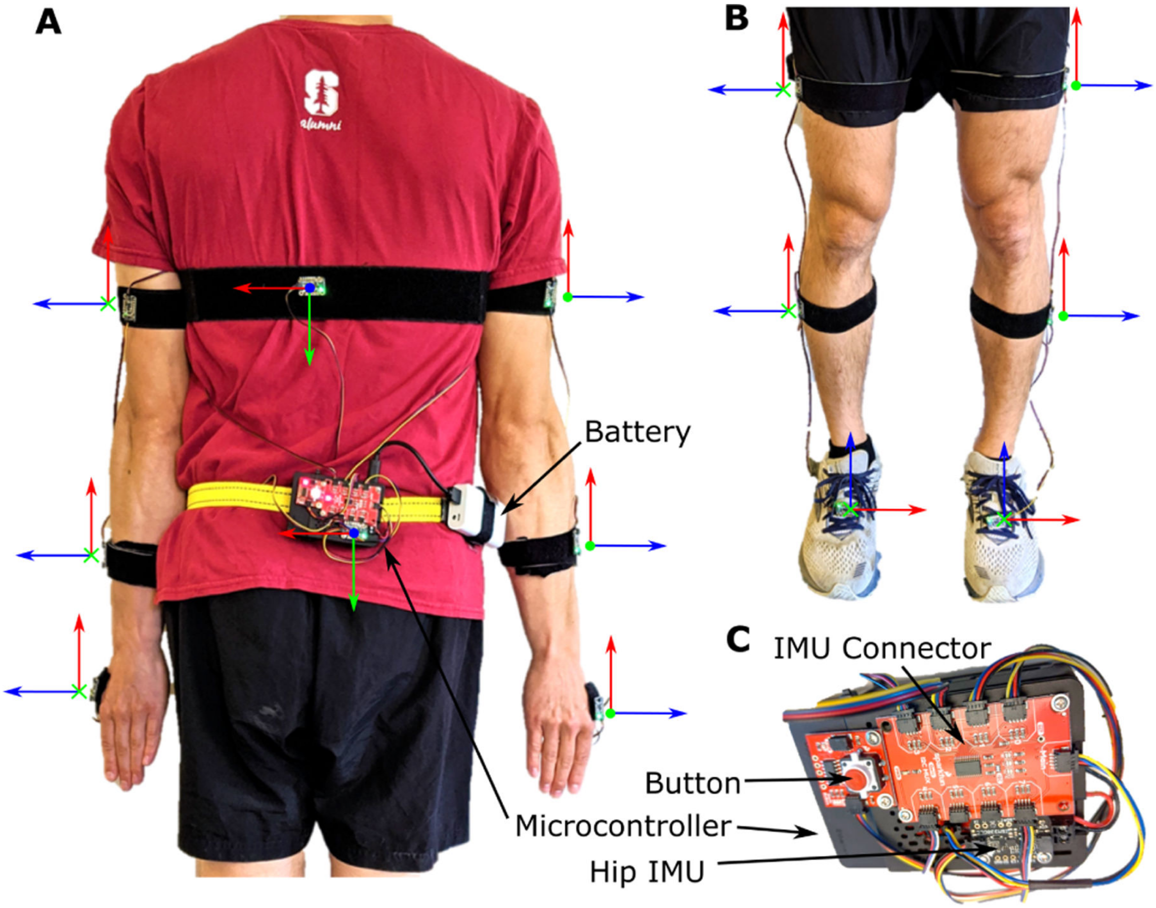

Fig. 1.

The components and default IMU orientations of the OpenSenseRT System. (A) An IMU on the pelvis is required and acts as the base in order to compute the relative orientation of other sensors. The OpenSenseRT System accommodates a variable number of additional IMUs to customize which kinematics are measured. To monitor movement of the upper body, three IMUs may be placed on each arm (on the upper arm, forearm, and hand). An additional IMU can be placed on the torso. The orientation frame with axes shown in red, green, and blue is used to orient the ***x***, ***y***, and ***z*** axes defined on each IMU. These individual body frames should align with the world reference frames of the fore-aft, mediolateral, and vertical axes, while the subject's joint segments are aligned in a neutral standing (or other known) position. (B) The lower-limb IMU placements also require the pelvis IMU as a base and can include up to three IMUs on the thigh, shank, and foot of each leg. OpenSenseRT allows for 1 to 14 of these IMUs to be sampled in a custom configuration. (C) A zoomed in view of the system components shows the microcontroller, battery, button (for starting and stopping recordings), IMU connector, and pelvis IMU.

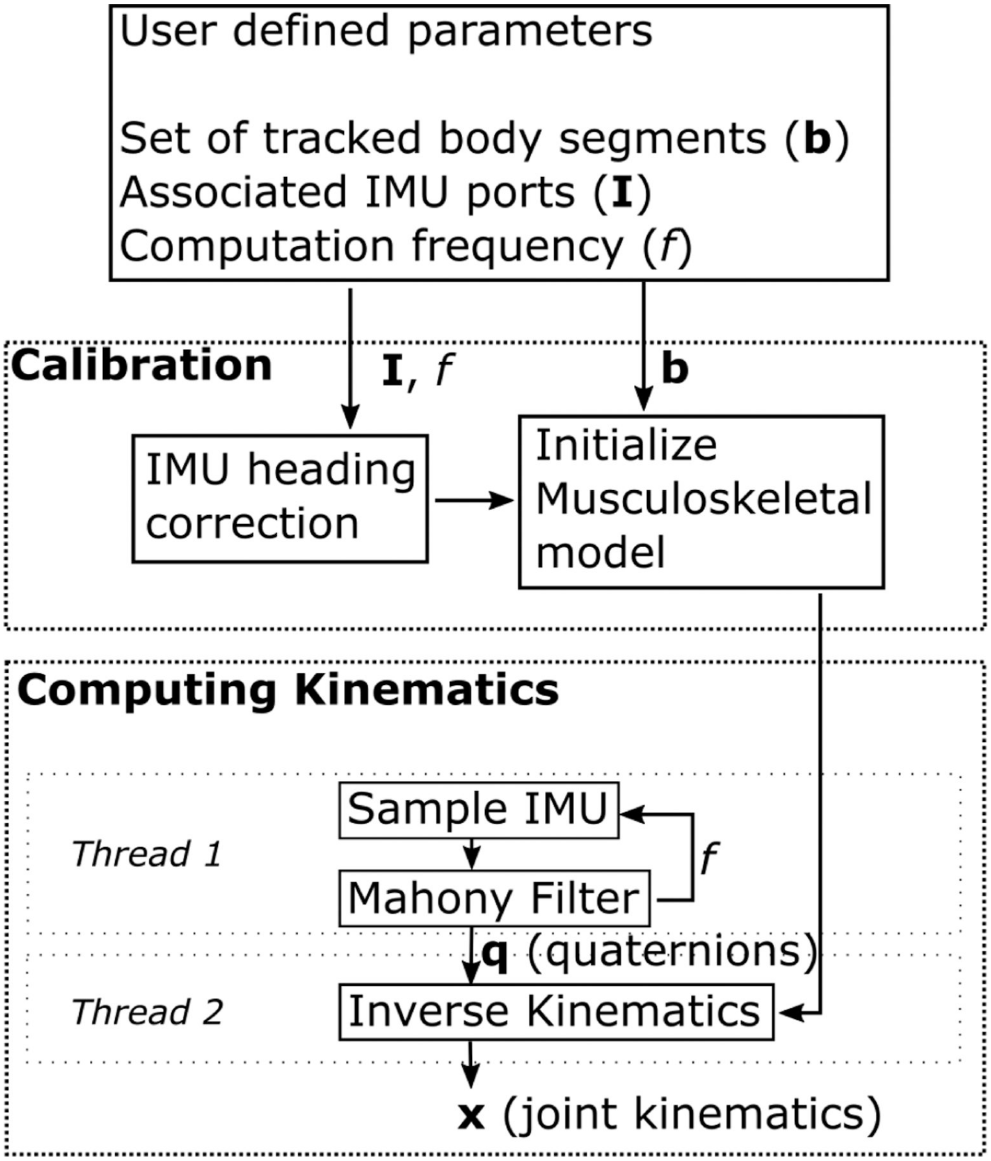

Fig. 2.

Flowchart for computing joint kinematics. The user defines the set of body segments to track, the IMU ports associated with the segments, and the computation frequency. The calibration process consists of a heading correction for the given IMUs and initializing the pose of the musculoskeletal model. The calibrated model and initial pose are passed to the inverse kinematics solver. One thread on the microcontroller records raw IMU data and computes orientations for each body segment using a Mahony Filter at each step. A second thread takes these orientations and the musculoskeletal model to solve the inverse kinematics, estimating joint kinematics at each time step.

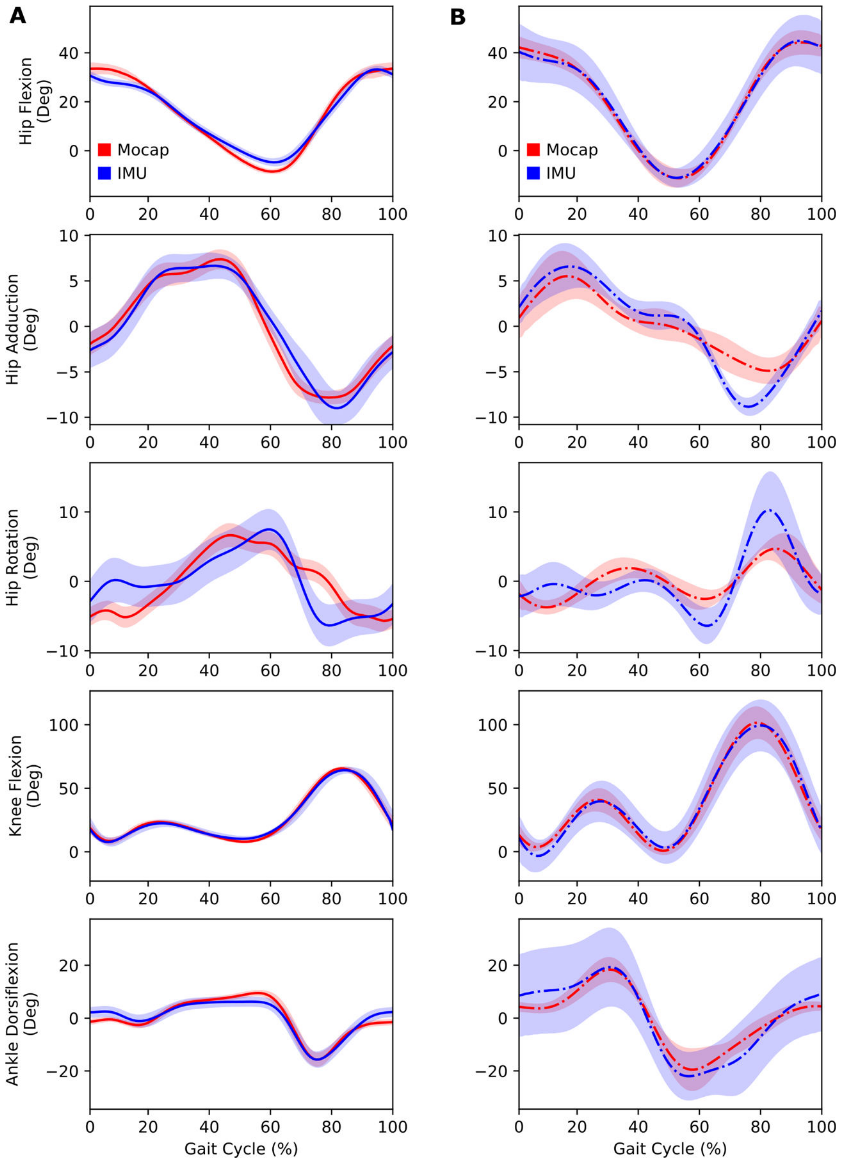

Fig. 3.

Lower-limb joint kinematics computed with the OpenSenseRT system compared to optical motion capture. (A) Joint kinematics during walking at 1.25 m/s were computed in real-time at 30 Hz with the IMU system for five subjects. The error bands represent one standard deviation of the RMSE across all subjects. (B) Joint kinematics during running at 4.0 m/s were computed offline (represented with the dash-dot line) at 100 Hz with the IMU system for a single subject. The error bands represent one standard deviation of the RMSE across all gait cycles for the single subject. Running requires a faster computation rate than the IMU system can perform consistently in real-time, although it can be analyzed in real-time over short bursts by using all cores.

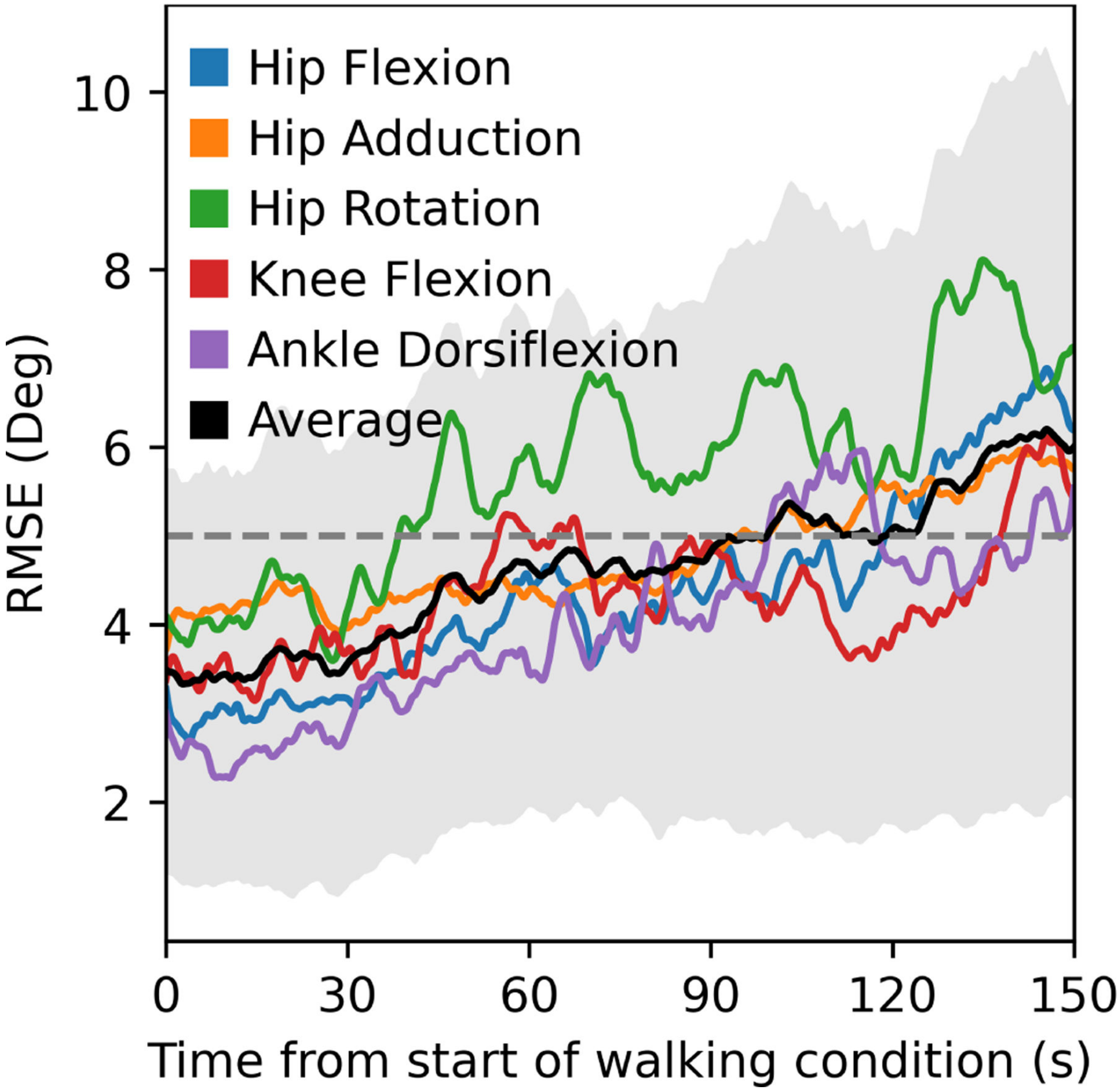

Fig. 4.

RMSEs for joint kinematics computed with the IMU system compared to optical motion capture over time for walking. The RMSE increased from approximately 3.5 to 6 degrees over the first 2.5 minutes of the condition. Hip rotation had the largest RMSE and drift. The black line and grey error band represent the average and standard deviation in RMSE across all subjects and joints.

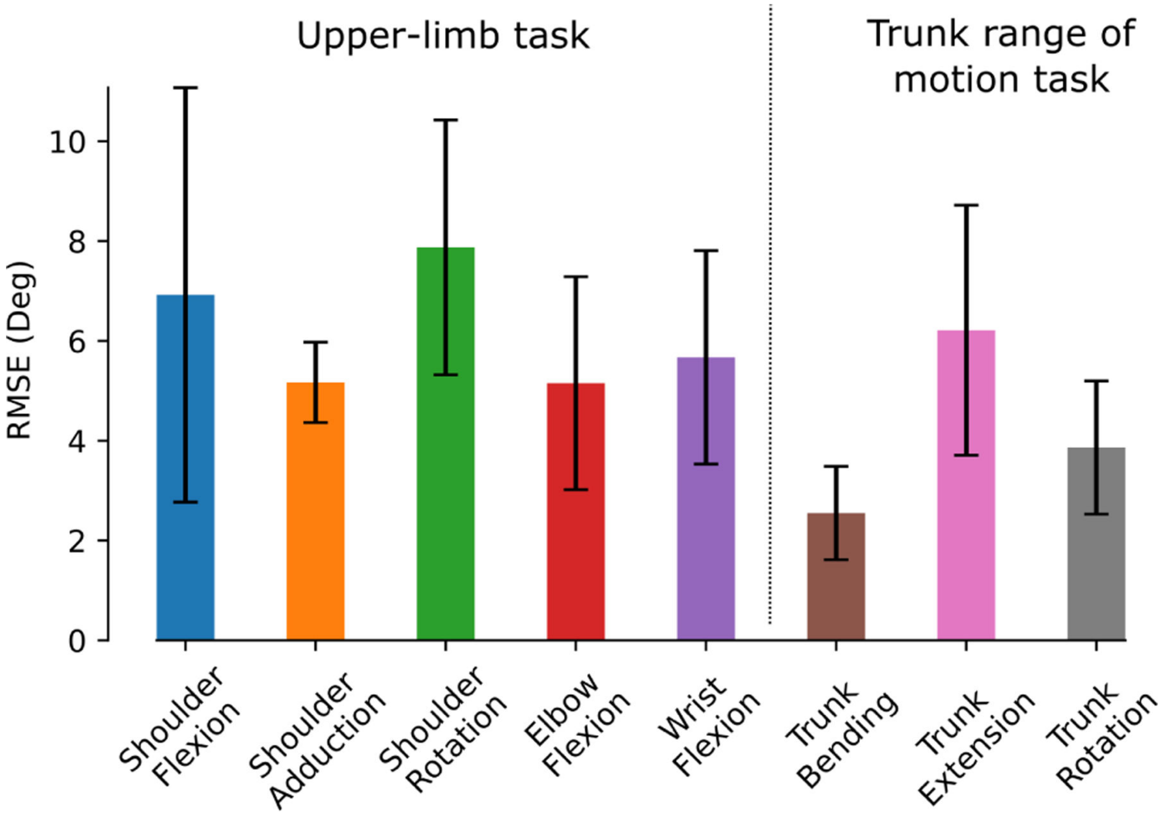

Fig. 5.

The average RMSE across each joint for the upper-limb task and the trunk range of motion task. The error bars represent one standard deviation of the RMSE across all subjects.

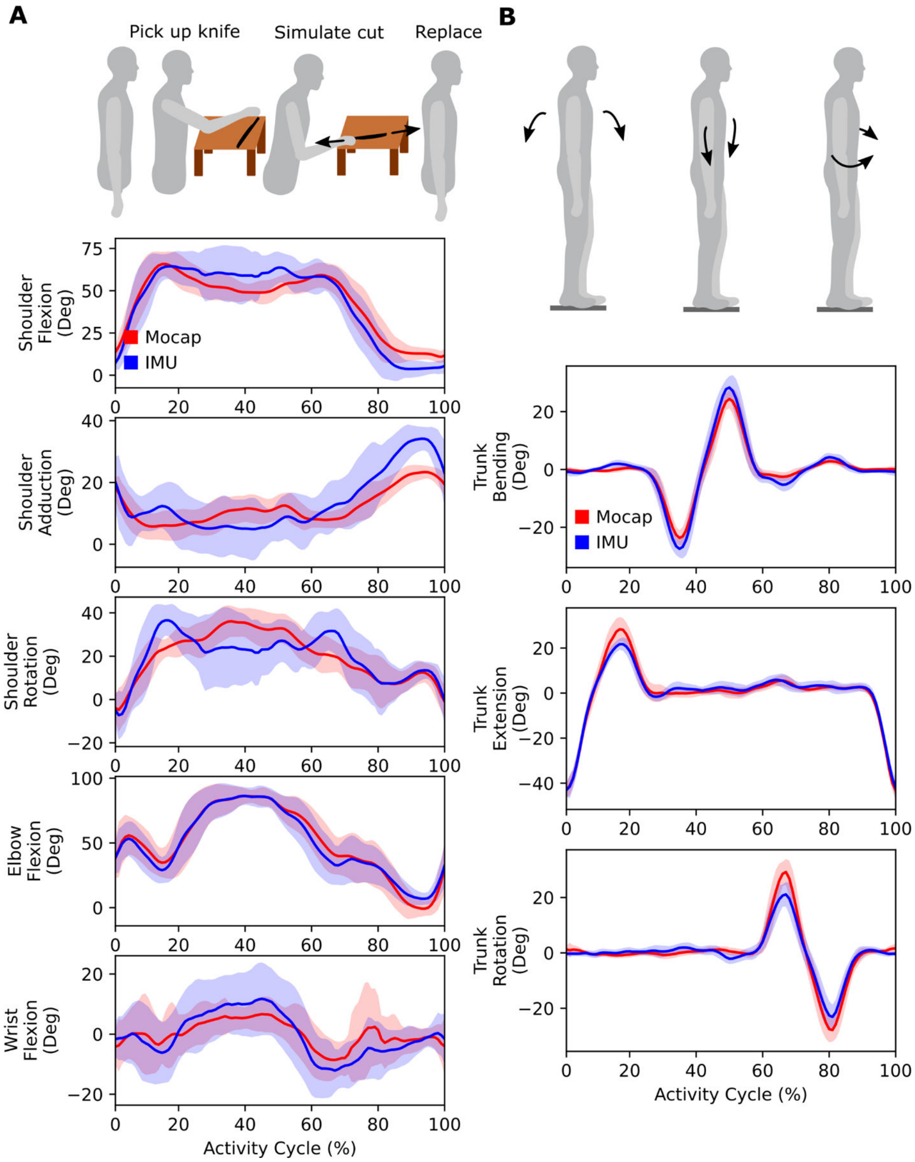

Fig. 6.

Upper-limb joint kinematics computed with the IMU system compared to optical motion capture. (A) An upper-limb task from the Fugl-Meyer assessment simulated cutting food. Subjects picked up a cutting utensil bent over a dish, and performed a cutting motion before placing the utensil back down. The task was repeated ten times to emulate an upper-limb "activity cycle" to look at average kinematics for one subject. (B) In the trunk range of motion task, subjects performed trunk flexion, bending, and then rotation. In both (A) and (B), the bands represent one standard deviation of the RMSE across all subjects.

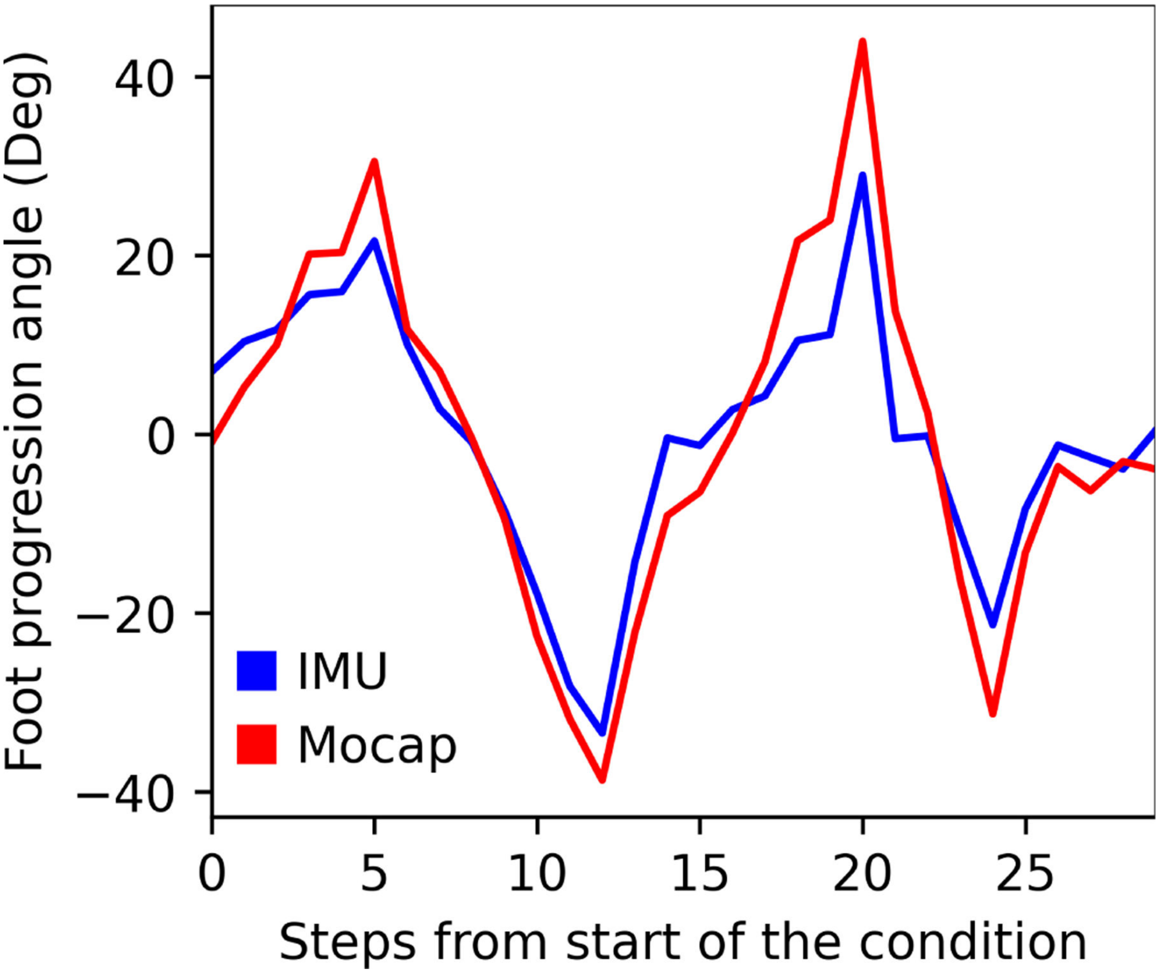

Fig. 7.

Comparison of foot progression angle computed by the IMU system and motion capture for one subject. The subject varied their foot progression angle with each step. The IMU system followed the same trends as the motion capture system but with smaller values across most steps. The average RMSE across all steps was 7.6 degrees.

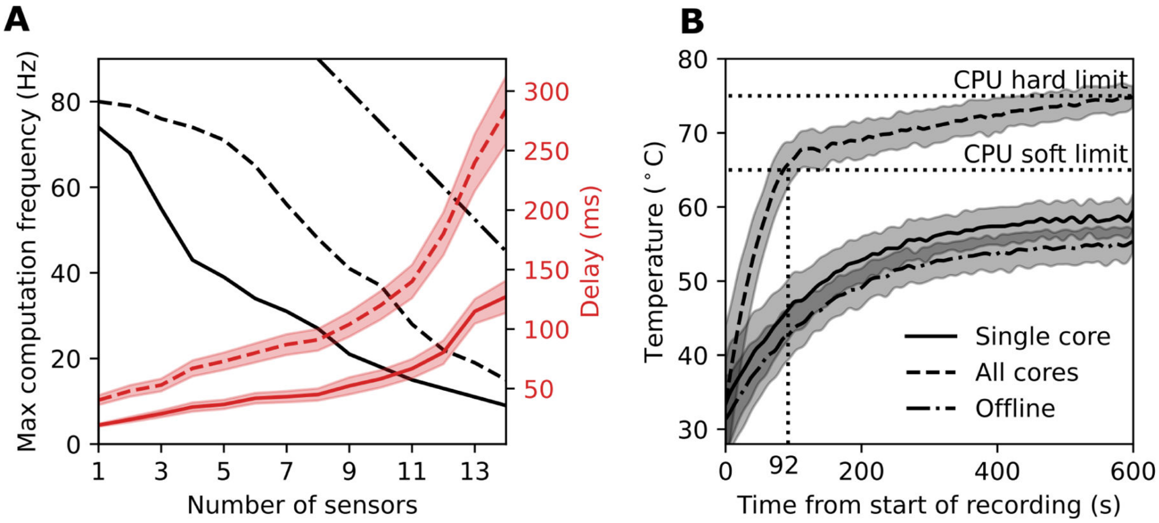

Fig. 8.

Computation and temperature limits of the microcontroller. (A) The maximum sampling rate and associated average delay depends on the number of sensors attached to the system and the mode of computation. We evaluated three modes of computation on the microcontroller: using a single core in real-time (solid line), all four cores in real-time (dashed line), and offline computation using a single core (dash-dot line). The sampling rates (black) and delays (red) were averaged over a one-minute recording. The error band represents one standard deviation across 5 trials. (B) The internal CPU processor temperature as a function of time was averaged for each of the 14 sensor configurations. The single core (solid line) and offline (dash-dot line) tests resulted in a safe steady-state temperature below any CPU limits.

TABLE I

Comparison of wearable motion capture systems that compute real-time kinematics

System	Number of sensors (*N*)	Computation frequency (Hz)	IMU	Cost (USD)	Validated kinematics (Y/N)	Portable Computation (Y/N)	Open-source (Y/N)	Battery life (hours)
OpenSenseRT	1–14	90–8	ISM330DHC	100+20**N*	Y	Y	Y	6.5
Kong, et al. [4]	6 (legs)	200	Custom	N/A	N	N	N	N/A
Zhou, et al. [12]	3 (arm)	100	Xsens M1b-9	N/A	Y	N	N	N/A
Xsens MVN Software [13]	1–15	60	Xsens MTw Awinda	32,000+	Y	N	N	6–8
Maruyama, et al. [14]	6 (legs)	60	Xsens MTw Awinda	12,000+	Y	N	N	6–8
Pedalvatar [5]	12	50	Custom	Laptop+100**N*	N	N	N	N/A
Miller, et al. [15]	2 (arm)	N/A	Custom	700	N	N	N	N/A
Andrews, et al. [16]	15	60	Custom	N/A	Y	N	N	N/A
Raghavendra, et al. [17]	3 (arm)	250	Custom	Laptop+100+55**N*	N	N	N	N/A
Zhang, et al. [18]	1–15	50	Custom	N/A	Y	N	N	N/A
Sy, et al. [19]	6 (legs)	100	Xsens Awinda Mtx	12,000+	Y	N	Y	6–8

TABLE II

Error and correlation between imu and motion capture lower-limb kinematics

Joint	Walking RMSE (Deg) Mean±Std	Walking Pearson Correlation Coefficient	Running RMSE (Deg) Mean±Std	Running Pearson Correlation Coefficient
Hip Flexion	4.5±2.3	0.995	8.4±5.9	0.999
Hip Adduction	4.8±1.2	0.981	3.4±3.2	0.964
Hip Rotation	5.2±2.2	0.758	4.1±3.8	0.782
Knee Flexion	4.3±0.7	0.997	7.1±5.4	0.994
Ankle Dorsiflexion	3.9±1.3	0.957	7.2±6.1	0.979

TABLE III

Error and correlation between imu and optical motion capture upper-limb kinematics

Joint	RMSE (Deg) Mean±Std	Pearson Correlation Coefficient
Shoulder Flexion	6.9±4.2	0.972
Shoulder Adduction	5.2±0.8	0.871
Shoulder Rotation	7.9±2.6	0.782
Elbow Flexion	5.2±2.1	0.987
Wrist Flexion	5.7±2.1	0.924
Trunk Bending	2.6±0.9	0.983
Trunk Extension	6.2±2.5	0.961
Trunk Rotation	3.9±1.3	0.949