MotionSynthesis Toolset (MoST): An Open Source Tool and Data Set for Human Motion Data Synthesis and Validation

Terrell R. Bennett, *Student Member, IEEE*, Hunter C. Massey, *Student Member, IEEE*, Jian Wu, *Student Member, IEEE*, Syed Ali Hasnain, *Student Member, IEEE*, and Roozbeh Jafari, *Senior Member, IEEE*

***Abstract*— Access to experimental data in the development of algorithms and techniques for wearable computing devices and body sensor networks allows faster validation and refinement of algorithms. The MotionSynthesis Toolset is an open source toolset and database built to assist in data collection and data sharing, and allow collaboration in review and validation of data sets. The tools can generate a sequence of movements and synthesize a data stream based on the data stored in the database. New movements can be added to the database and the tools by the community. The tools also allow visualization, and validation of the movements and data with video and signal waveforms. The data set has more than 20 subjects and multiple repetitions of the movements from each subject to increase data diversity.**

***Index Terms*— Wearable computers, open source, databases, data synthesis, body sensor networks.**

I. Introduction

WEARABLE computing devices and body sensor networks (BSNs) are becoming more commonplace in and out of research environments. They are being used for health monitoring [1], [2], activity tracking [3], [4], and fitness applications [5]. Developing new algorithms and techniques to detect daily activities more efficiently and with higher precision is an important area of research. Validation of new systems and approaches is a critical part of research and development. The validation process typically requires the design of experiments and the collection of new data, which take time and make validation more time consuming. Access to datasets captured from wearable motion sensors can be useful as they can enable the validation of these algorithms and techniques without the need for new experiments or data collection. Furthermore, if one researcher collects data to test an algorithm and another researcher collects data to test another algorithm, it can be difficult to compare these algorithms fairly because of the differences in the data. Using a single dataset, like MoST, allows for fair comparisons between algorithms and for repeatable tests with adjustments and updates to algorithms.

Activity monitoring and motion sensing with wearable computers is attracting more research attention due to the low cost, easy set-up and ubiquitous sensing ability. Various research topics are widely studied for wearable motion sensor based activity monitoring systems that target optimizing power consumption, increasing activity recognition accuracy, and selecting the best sensor subsets. All of these topics can be better understood by having a complete and robust dataset. Thus, data is of crucial importance to design, optimize, and validate algorithms; and a dataset like MoST can accelerate research by reducing the time and effort related to the collection and validation of data.

Collection and validation of large amounts of data can be difficult for multiple reasons. There may be technical limitations and difficulties in capturing certain scenarios [6]. Time is a major issue for collecting large amounts of data. Setting up equipment, attaching sensors to the subject, and collecting data can all take a significant amount of time. A large number of subjects is also, typically, required to generate a large amount of data. Data collection time must be scheduled with each subject, which can be difficult depending on the number and types of subjects and their availability. Validation of the data is normally handled by the group that collects the data. Methods such as self-reporting by the subject, video recording of the collection, and data inspection are used to validate the data. These methods are effective, but mistakes can be made and the process can be very labor intensive.

Another consideration to be made when collecting large volumes of data is the storage and characterization of the data. Meta-data, with information about the collection process and the origins of the data, can be very important for users. Additionally, file storage (*e.g.* text files, databases, etc.) will determine the accessibility of the data for users.

Manuscript received March 4, 2016; revised April 22, 2016; accepted April 28, 2016. Date of publication May 4, 2016; date of current version June 2, 2016. This work was supported in part by the National Science Foundation under Grant CNS-1012975 and Grant CNS-1150079, in part by the TerraSwarm Research Center, one of the six centers of STARnet, through a Semiconductor Research Corporation Program within Microelectronics Advanced Research Corporation and Defense Advanced Research Projects Agency, and in part by the National Institute of Health under Grant R15AG037971. The associate editor coordinating the review of this paper and approving it for publication was Prof. Janice Limson.

T. R. Bennett and H. C. Massey are with the University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: tbennett@utdallas.edu; hunter.massey@utdallas.edu).

J. Wu and S. A. Hasnain are with Texas A&M University, College Station, TX 77840 USA (e-mail: jian.wu@tamu.edu; syedali.hasnain@tamu.edu).

R. Jafari is with the Center for Remote Health Technologies and Systems, Texas A&M University, College Station, TX 77840 USA (e-mail: rjafari@tamu.edu).

Digital Object Identifier 10.1109/JSEN.2016.2562599

To reduce some of the difficulty in data collection, validation, and generation, we presented the MotionSynthesis Toolset (MoST) [6]. This toolset allows the generation of a sequence of movements, synthesis of a data stream using real data, visualization and validation of the sequence of movements and data through the use of video and waveforms. While the initial version of this tool resolves some individual data collection difficulties, it was a closed system. The development team would be responsible for all data generation and tool maintenance.

Open Source development concepts have been used in the software community to quickly improve the capabilities as well as the quality of software tools. Allowing other groups to contribute to the tools, increase the amount of data, and assist with data validation will allow MoST to improve more rapidly. In addition, the amount of data that the tools can use will increase at a much faster rate than the closed model. A large and broad database will allow for more diverse datasets to be synthesized by the tools, which will help meet the needs of a larger section of users.

MoST can quickly provide data that can speed up algorithm development for wearable sensors. Researchers that do not have hardware to conduct their own experiments can generate real sensor data that can be used to verify their algorithms. Researchers with their own sensors can use the tool to gain access to new data and add their own data to the database. Making updates to the tools for ease of use for data access as well as modification by the community is critical to achieve this goal.

The primary contributions of this paper are:

1. A movement database with up to 15 repetitions for each of 23 movements from more than 20 subjects (increased from four in prior work [6]) with six sensor locations
2. Tools that allows a sequence of movements (diary) to be determined and synthesized based on data in the dataset
3. A tool to visualize and verify the data from the sensors as well as the related video for the movements
4. An online search engine to find and download data from the database based on movement and subject metadata (*e.g.* age, sex, height)
5. We validate the uses our dataset and our data synthesis tools through experiments using a pattern matching algorithm

The remainder of this paper is organized as follows. Section II of the paper covers related works. Section III describes our data collection process, hardware, and software. Section IV covers the MotionSynthesis tools and the related updates. We present a validation of our synthesis and use of the data for algorithm development in Section V. Finally, we demonstrate another dataset working with MoST and discuss plans for further enhancement of the tools in Section VI and our conclusions Section VII.

II. Related Works

Vision based datasets were some of the first available and widely used for human activity recognition and pattern recognition. While these datasets enable users to develop and validate their algorithms rapidly without putting a lot of effort into collecting and annotating data, the datasets only enable users to compare their algorithms based on the original data in the dataset. A video dataset is published for six different actions for 25 subjects in four different scenarios, which are used for action recognition [7]. A similar dataset is proposed in that includes 10 classes and 90 videos for outdoor actions [8]. Video datasets are extracted from television programs [9] and movies [10]. The datasets include sports activities and Hollywood movie actions respectively. Datasets with a large number of classes and videos are proposed [11], [12]. The datasets mentioned above are static and the data are meant solely for use as they were captured. Also, vision based motion datasets are limited for ubiquitous sensing applications because of the line-of-sight problem. In essence, the subject must always be in view of the camera.

The activities of daily living can be divided into two primary categories. The first type of activity is the low level activity, which includes the postures (*e.g.* sitting and standing), the short-term transition activities (*e.g.* sit-to-stand and sit-to-lie) and the dynamic periodic activities (*e.g.* walking and running). The second type is high level activity, which is characterized as long-term complicated tasks that can be composed from the low level activities (*e.g.* watching TV and eating dinner). For example, eating dinner can be composed by the low level activities of sitting, eating, and drinking.

Several datasets have been published for wearable motion sensor based activity monitoring systems which are focused on low level activities. Wearable Action Recognition Database (WARD) is published by UC-Berkeley and it covers 7 female and 13 male subjects, with age ranging from 19 to 75 [13], [14]. Each sensor mote has a 3-axis accelerometer and a 2-axis gyroscope and 5 sensors are attached to different body parts. A dataset was offered by the University of Rio de Janeiro that covers 5 activities of daily living from 4 healthy subjects with 3-axis accelerometers on 4 different body segments [15]. Similarly, the database USC-HAD was offered that includes 7 female and 7 male subjects with different height, age and weight [16]. Only one waist motion sensor (3-axis accelerometer and 3-axis gyroscope) was used to capture 13 activities of daily living. The NINAPRO database looks at finer hand and wrist movements using sEMG sensors [17]. Although these datasets offer various low level activities of daily living with different data collection configurations, it is hard for the user to add more low level activities to the database in order to benefit a wide range of research interests. The user may want to investigate different activities or more activities than the dataset. Our toolset has software and an interface that enables the user to annotate and add more activities to the database in a straightforward manner. More significantly, our tool is able to synthesize a high level activity or a sequence of low level activities that is of interest to the user. This data synthesis feature enables researchers to synthesize activities instead of collecting data for different scenarios. This can reduce the cost of data collection while accelerating the design and validation tasks significantly by using data that has been validated.

Several other datasets were released for the high level activities. The TNT15 dataset captures video data from

8 RGB cameras along with 10 IMU sensors. For 5 movements [18]. A database was offered to cover 17 high level activities of daily living for 8 different scenarios [19]. This database provides an interface to annotate and display the low level activities which compose the high level tasks. The MIT PlaceLab dataset provides a variety of high level activities of daily living captured from a fully instrumented apartment with various types of environmental sensors [20]. CMU-MMAC, CMU Multimodal Activity Database, was published to provide human activities in the kitchen, including cooking and food preparation [21]. Four different modalities, including motion capture, audio, video and inertial sensors, were used for the data capture. Another database was released for high level activities with information of 72 sensors of 10 modalities [22]. Unlike these datasets, our database only uses inertial sensors which are focused on activity monitoring using the motion sensor modalities. All of these datasets are static and do not offer tools with the ability to synthesize different high level activities outside of the existing dataset. Another advantage of our dataset and tools is the ability to generate a large amount of data with both sensor data and video.

Data management for wearable computers has been well studied. However, the existing works focus on techniques that involve efficient management of data repositories and enhancing the speed of recognition and query processing [23]–[28]. The importance of offering data for design and validation is ignored in these works. An open source set of tools for context recognition and data validation was created [29]. This toolchain assists in data collection and validation. Though it is open source, it has not been updated recently and does not have a method for outside users to add new data. Additionally, it focuses on many sensors and a long duration. Our dataset provides segmentation for low-level activities (*e.g.* sit-to-stand, sit-to-lie, etc.) based on likely locations for wearables.

The proposed platform offers many of the capabilities of the existing datasets, and it enables users to synthesize data streams for a subject's activities and scenarios of their choice. This feature has the potential to accelerate the design, prototyping and validation efforts. It can reduce the time associated with system development and potentially provides more extensive data for validation and system refinement. Additionally, the open source nature allows the community to add data to the repository and functionality to the toolset.

III. Data Collection

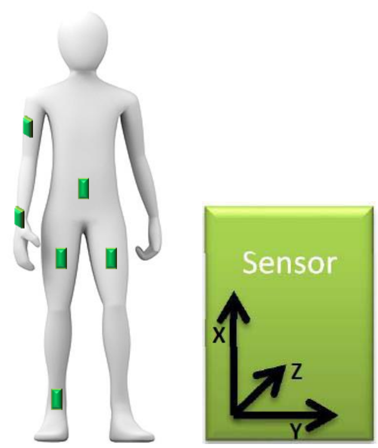
As described in our prior work, data collection is one of the critical tasks in the process of creating a database and related tools for data synthesis [6]. For this work, data collection refers to capturing a large amount of data for low level human activities from different subjects using inertial measurement unit (IMU) based sensors. A consistent collection process that covers a basic set of movements along with hardware and software tools is used to ensure the data collected is reliable and that it can be validated.

A. Process

The activities selected for the MoST database were chosen to be useful for multiple types of research activities. A set of the most commonly used low level activities of daily living [30] were selected. This set of movements is common for a variety of applications and would therefore be useful for a large community of users. Each activity is associated with a beginning and ending posture (*e.g.* sitting, standing, etc.), and each posture has at least one movement in the dataset that transitions into and out of that posture. This requirement is in place to ensure compatibility with the developed tools. The initial set of movements collected for the MoST database can be seen in Table 1 ordered by the number under which the movement is stored.

TABLE I
Movements in the Initial MoST Database

1.	Sit to Stand	2.	Stand to Sit
3.	Sit to Lie	4.	Lie to Sit
5.	Step Forward and Backward	6.	Looking Back Right
7.	Grasp from Floor	8.	Turn Right 90 Degrees
9.	Grasp from Shelf	10.	Jumping
11.	Step Left then Right	12.	Eating
13.	Drinking	14.	Stand in neutral posture
15.	Sit w/feet on floor	16.	Basic Lying
17.	Walking	18.	Sit w/leg crossed
19.	Sit w/ankles crossed	20.	Stand w/leg crossed
21.	Stand w/one leg forward	22.	Kneeling
23.	Use a Phone		



(a) (b)

Fig. 1. Model showing the sensor locations used for the data collections for MoST as well as the sensor orientation and axes. The locations are possible wearable devices and common locations used for research applications.

For consistency and to properly capture the set of movements described above, six sensors were attached on the subjects for each collection. Specifically, the body parts on which the sensors were deployed, are 1) the right ankle, 2) the waist, 3) the right arm, 4) the right wrist, 5) the left thigh, and 6) the right thigh as shown in Fig. 1 (a). This configuration serves two other purposes. For others that use the data, the sensors are located in the most common locations for commercial and research based wearable devices. This ensures that users will be able to get data for target locations that a

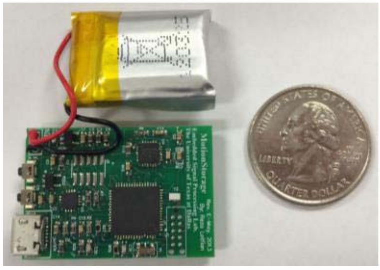


Fig. 2. Custom sensor board with 3-axis accelerometer, 3-axis gyroscope, and TI MSP430 used for data collection.

sensor will likely be worn. Secondly, as others add data to the database, these sensor locations will likely provide overlap that will more easily allow combining and comparing data from multiple sources in the tool. Fig. 1 (b) shows the orientation of the sensor axes when the subject is in a neutral standing position (*i.e.* the position shown in Fig. 1 (a)). Acceleration is measured along each axis while rotations are measured about the axis of rotation. All data stored in the database will use acceleration and rotation data relative to the sensor's reference frame.

This sensor configuration was used to ensure all major limbs could be monitored and that the movements of interest could be accurately captured by the sensors. It may have been useful to put a sensor on both sides of the body and at every rigid link; but a Bluetooth network supports a maximum of seven slave devices at a time. Additionally, we expect the movements from one side of the body to be mirrored on the other. These factors led us to work, primarily, with a single side of the body. Some sensors may be of little significance for some movements, but all sensor information is recorded for the entire set of movements consistently. The availability of all sensor data can be helpful for the research topic of subset sensor selection.

MoST users must recognize that the open source nature of the tool means that the specifics of the data collected may vary. The number and the location of sensors may be altered depending on the group doing the collection. Additionally, the populations that the research groups have access to will likely change the types of movements and data collected. This should be an advantage for researchers looking for large amounts of varying data, but it could create inconsistencies in the sensor positions and the quality of the data collected.

We have increased the number of subjects in the MoST database from 4 subjects in our prior work [6] to more than 20 subjects. Each subject performs each movement 15 times consecutively with a short pause between repetitions. This repetition and the increase in the number of subjects adds greater variety and diversity to the dataset. Another important factor in the data collection process and the open source nature of MoST is the ability of a community of users to view the raw data, ask questions, make corrections, or apply new annotations to the database.

B. Hardware

Fig. 2 shows our lab-developed 9-axis wearable motion sensor that is used for the data collection. It consists of a Texas Instruments MSP430 microcontroller, a dual mode Bluetooth module, and an InvenSense MPU9150 which has a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis magnetometer. The sampling rate is set to 200 Hz which is relatively high for the movements we collected, but ensures that users will have enough resolution for any signal processing. Additionally, data can be decimated if this sampling rate is too high for a specific user. The accelerometers were configured with a range of $\pm$2g, and the gyroscopes were set with a range of $\pm 250^\circ/second$ based on the low intensity of our initial activity set. The sensor configurations are provided along with the captured data. It is difficult to guarantee a homogenous magnetic field and this information would vary outside of the environment in which it was captured. Therefore, though the magnetometer data is collected, only the accelerometer and gyroscope data are reliable and only these modalities should be used.

The motion data is captured with 16 bit analog to digital converters (ADCs) on the MPU9150 and passed through an I^2C interface to the microcontroller. Each sample is time stamped and packetized and passed to the Bluetooth module via UART. The data is then transmitted to a PC and logged into a text file. At the same time, synchronized video is recorded with a web camera connected to the same PC at 15 frames per second (FPS) with 720p resolution. The synchronized camera information allows visual validation and annotation (*i.e.* parsing) of the movements.

C. Software

Two software tools were designed and developed for the data collection and validation. The first tool is the data collection software that provides the ability to set the configuration parameters before starting the collection and logs all the motion data into text files and a related video file at the same time. The file format of each sample is as follows: AccelX, AccelY, AccelZ, GyroX, GyroY, GyroZ, MagX, MagY, MagZ, SensorPkt#, SensorTime, PCPkt#, PCTime. The first three columns are accelerations and the second three are the angular velocities. The following three columns are magnetometer data. The next two columns represent the packet number and time stamp from sensor. The final columns are the PC packet number and time stamp. Additionally, each video frame is time stamped to ensure accuracy and synchronicity in the case of inconsistent frame rates.

Each subject performed each movement 15 times and the data is parsed by individual repetitions for the database. Using the Data Visualization tool discussed in Section IV-B, the start and end of the movements are annotated using the video as the ground truth. After this, the sensor data and the corresponding videos are parsed and stored in the repository. Due to feedback from users of the original MoST data from our prior work, the data collected is converted to and also stored in the HDF5 [31] hierarchical storage format. This format allows for files that contain sensor data as well as meta-data for the collections in the database. These new data files and video clips represent the building blocks for the database that is used in the MotionSynthesis Toolset.

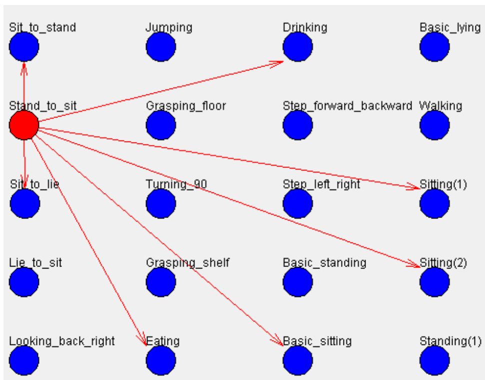


Fig. 3. The Graph Panel Display shows the available movements and which movements can follow the selected movements.

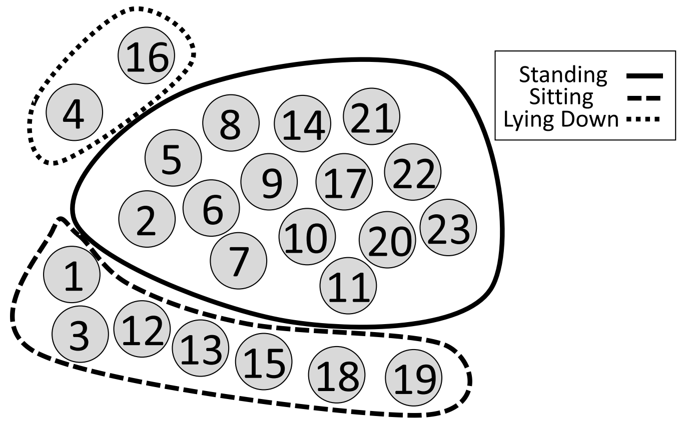


Fig. 4. Hypergraph for the starting posture of the movements in the database.

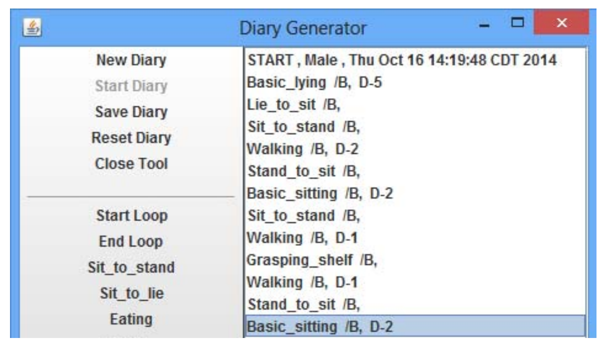


Fig. 5. Diary Generator tool which allows the user to input a sequence of movements and options based on the database.

IV. Motion Synthesis Tools

A. Diary Generation Tools

The Diary Generation tools include the Diary Generator and the Graph Panel. These are the primary input tools in this toolset and the key to generating the desired data streams. They are Java based tools with a graphical user interface (GUI) that allows users to select the relevant aspects of the motion diary, view the movements and movement relationships, and add or remove movements from a diary.

1) Graph Panel: The Graph Panel is a directed graph that shows all available movements as vertices. Directed edges connect selected movements to possible subsequent movements. Fig. 3 shows the Graph Panel with a single movement, stand-to-sit, selected. The seven movements that can be done after this movement are connected in the figure. In the prior version of this tool, movement pairs had to be created for each movement. These pairs included each movement that could occur before the new movement followed by the new movement, and the new movement followed by each movement that could occur afterward. As the number of movements in the system increases, this would become increasingly cumbersome.

The Graph Panel tool was updated to alleviate this difficulty by using hypergraphs [32] to determine the movement progressions and for simplicity in adding new movements. A hypergraph is a graph in which each edge can connect multiple nodes. There is a hypergraph for starting postures and another for ending postures. The ending posture of the selected movement is compared to the hypergraph for starting postures to determine the next available movements. For example, if a stand-to-sit occurs, the ending posture is sitting. Based on this posture, all movements that have sitting as a starting posture are available for the next movement. Each edge of the hypergraphs represents a starting or ending posture for the movements. Fig. 4 shows the hypergraph for the starting postures based on the 23 movements in the initial database using the numbering from Table 1. The nodes, represented as circles are the movements. The edges of the graph are the starting postures for the movements.

Using the hypergraphs allows users to easily add new movements to the tool and allows the tool to seamlessly handle the movements and movement progressions without requiring knowledge of any previously existing movements or the related postures of those movements. The starting posture, ending posture, and movement name are the only pieces of information necessary to add a new movement to the diary generation tool. The tool handles adding the new nodes to the proper edges in the hypergraphs and the Graph Panel display. For example, adding a movement to MoST that starts and ends with a standing posture (*e.g.* running) would require 28 entries in the original tool vs a single entry in the updated tool. This greatly reduces the level of effort for adding new movements compared to the prior version of the tool and makes additions by the community easier.

2) Diary Generator: The Diary Generator tool allows the user to generate a diary which is a list of movements that will be used to synthesize data. When creating the diary, the user can choose from the six sensor locations used in our collections, from the different sensor modalities (*i.e.*, the gyroscope and accelerometer axes), and the subject from which data is to be generated.

The GUI of the Diary Generator tool is relatively straight forward: a sidebar contains options and the list of available movements, while the main panel contains the current diary information. This interface is shown in Fig. 5, which shows a small sample diary. One of the options available is a loop control, which is contained in the same segment of the sidebar as the movements. This allows the user to repeat a section of their diary a specified number of times.

```
START , M1 , Fri Feb 21 22:46:21 CST 2014

Sit_to_stand   /B,
Walking   /B,
Stand_to_sit   /REx,
STOP!  * 0:0{Right Ankle, Right Thigh, } <Acc X, >
```

Fig. 6. Diary file based on a sequence of movements.

The movement list in the sidebar will always contain only movements that can currently be performed. This refers back to the hypergraph constructed earlier. A movement cannot be performed if the subject is not currently in the posture necessary to start it (*e.g.*, a subject cannot lie down and then jump immediately after).

B. Data Synthesis and Visualization

The Data Synthesis tool allows users to generate new streams of data based on the diary generated. This can save significant time in testing algorithms based on a specific sequence of movements. After the user creates a diary file, data can be generated using the Data Synthesis MATLAB® tool. This tool processes the output from the diary generator and produces the associated data by selecting movements from the database based on the diary, and stitching together the individual movements to create the final output data. In the example input shown in Fig. 6, the x-axis of the accelerometer for the right ankle and right thigh sensors will be generated.

One of the important considerations for this operation is how the individual movement data is stitched together. In the original tool, this method was a simple concatenation of the segments. This has since been updated to include a smoothing filter in this process to provide more natural transitions and reduce the risk of any discontinuities in the data stream. The smoothing filter is a simple moving average filter with a span of 10 samples, performed on the 25 samples before and after each transition.

The main output of this tool is a set of files with data for each sensor in the same format as the input, which is described in Section III.C. This makes the output similar to a normal data collection, with generated packet numbers and timestamps. This tool also produces two other files, which are a list of the times at which a transition occurs (*e.g.* annotations) and the sequence in which the movements are performed. Both of these outputs are utilized by the Data Visualization Tool.

A screenshot of the Data Visualization Tool is shown in Fig. 7. This tool allows users to view their generated data as well as any associated video that goes with it, using the output from the Data Synthesis tool and the videos in the MoST database. For each movement in the diary, the appropriate video file is loaded and displayed for the user, and each transition between movements is clearly marked with vertical lines. More detail on the Data Visualization Tool is giving in our prior work [6].

C. MoST Database

Many of the existing databases provide a mechanism to access the original data files. Users may prefer access to the raw data files with no filtering, synthesis, or other modification. To ensure that this data is easily accessible to the community without having to access the other tools, the MoST Database search engine was developed and added to the original MoST toolset. The database stores the raw text files from all collections for MoST as well as other datasets. The search functionality allows the user to determine the characteristics of the data they would like to see (*e.g.* age of subject, sex of subject), select the data, and directly download the raw data files.

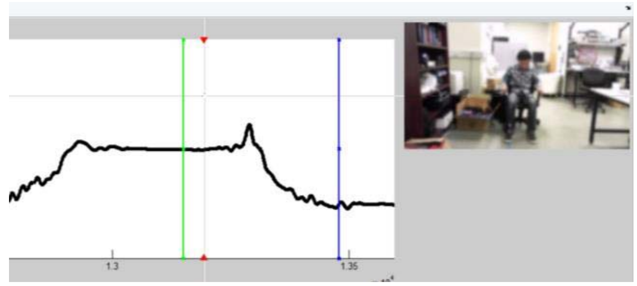


Fig. 7. Data Visualization Tool.

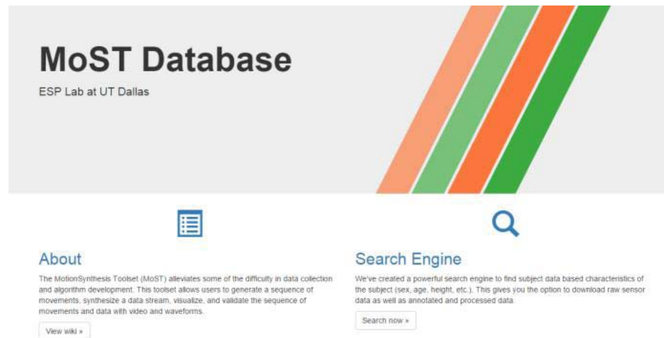


Fig. 8. Web based search engine for MoST database.

Fig. 8 shows the front end of the MoST database search tool. Unlike the other tools in MoST, the database tool is completely web based. This allows those that only want access to the raw data in a simple text format to get the files in a straightforward manner. Beyond the raw data, the users also have the option to download the annotation file that gives information on the starting and ending positions of the specific movement data in the raw data files.

Finally, users can use this web based interface to directly upload data to the web server. New uploads must be formatted and approved, but this capability reduces the difficulty in making new data searchable by web users.

V. Validation of MoST Data

A. Synthesis Validation

MoST offers the ability to generate sequences of movements based on the segmented single low level movements in the database. The synthesized sequences may not be the same as the natural sequence performed by a human. This difference may come from several factors: 1) there may be different motions when different subjects are performing the same movement, 2) data in a single sequence could come from multiple subjects in the database, 3) the stitching together of the movements could differ from natural transitions, 4) the slightly different orientations of the sensors might cause a small discontinuity in the transitions between movements. The data from any specific synthesis might not match a specific individual perfectly, but we want to show that data generated by the tool is similar to human data. Specifically, we want to show that synthesized data is no more different from a human than humans might be from each other. To validate the similarity of synthesized data streams and human subject's data streams, we will investigate an activity segmentation application.

We collected two sequences of movement data from five human subjects that also have data in the MoST database. During the collection of the validation sequences, we used the same sensor configuration that was used for the data collection. We synthesized the same sequences of movements using MoST for each of the validation subjects from their segmented data in the database. Therefore, we have natural sequences as well as synthesized sequences from the MoST tools for each validation subject.

To compare the synthesized and natural data, we trained templates based on the movements in the sequences for each of the subjects. Each movement was performed five times. Based on inspection of the data from the sensors and the video using the Data Visualization tool, we determined the sensor and modality (*i.e.* accelerometer or gyroscope) that best identifies the movement. Using dynamic time warping (DTW) [33], we compared the five instances of the movement to determine the most representative template from each subject. The DTW algorithm is used to measure similarity between two time series. The advantage of this algorithm is that it allows the speed or timing of the signals to vary. Therefore, if the same movement occurs at different speeds, the instances can still be compared and found to be similar.

In the algorithm, a template of a movement is compared to a stream of data to determine if the movement that the template represents occurs in the stream. The samples from the template are compared against the samples in the stream and the similarity between the template and the stream is called the DTW distance or DTW cost. Using this DTW cost as our metric, we compare to a pre-determined threshold to decide if the template's movement exists in the stream.

To determine the most representative template, we calculate the DTW distance between all pairs of templates and determine which instance has the smallest distance to all other instances. This instance becomes the representative template for each movement for each subject.

Table 2 lists the two sequences used for synthesis validation. The first sequence of movements represents a common set of movements for a student working in a lab. The second sequence represents part of a possible morning routine.

We use DTW distance and a threshold to detect the movements. The DTW algorithm used can also provide the beginning and ending of the movements in the stream. Using the set of templates from each subject, we identify and segment the movements from all data streams (synthesized and natural). The detection as well as the beginning and the ending of the movements are compared to annotations to determine a segmentation error. The natural sequences are annotated based on video while the synthesized data uses annotations generated by the Data Synthesis tool.

TABLE II
VALIDATION SEQUENCES

	Sequence 1	**Sequence 2**
Movement 1	Sitting	Lying
Movement 2	Sit to Stand	Lie to Sit
Movement 3	Walking	Sit to Stand
Movement 4	Grasping from shelf	Kneeling
Movement 5	Walking	Walking
Movement 6	Stand to Sit	

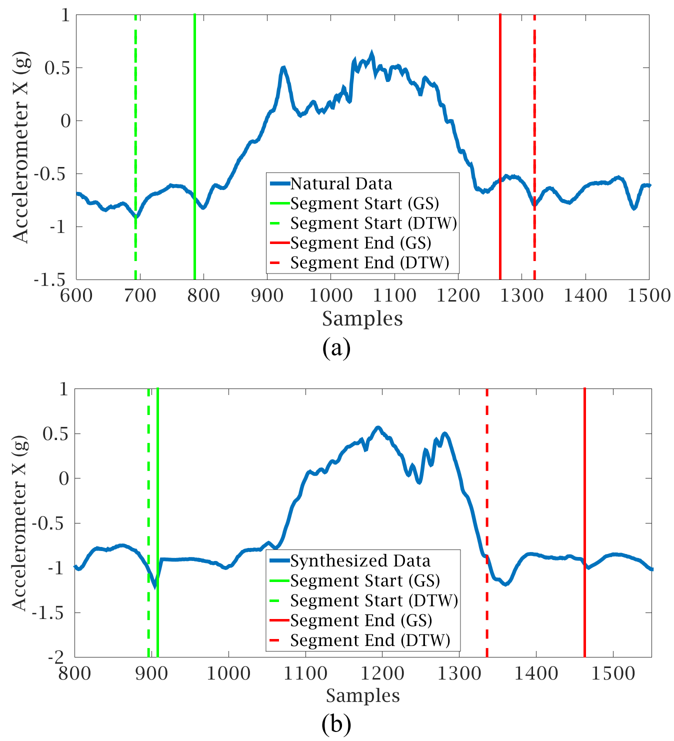


Fig. 9. Segmentation of movements for natural data (a) and synthesized data (b) from subject 3 for the grasp from shelf movement on the arm sensor. The gold standard is based on visual annotation (natural) or tool annotation (synthesized).

Fig. 9 shows the segmentation for a natural dataset (a) and a synthesized dataset (b) from the arm sensor for subject 3 performing the grasp from shelf movement. The solid lines represent the gold standard segmentation of the movements. We used video of the sequences to annotate the natural data sequences. The MoST tool output the annotations used for the synthesized data. The DTW algorithm is used to determine the start and finish of each movement. This algorithm output is used to create the vertical dashed lines that represent the algorithm segmentation. We find the difference between these values to determine the segmentation error reported in Table 3. Because the actual movements in both natural and synthesized streams are from real data, understanding the transitions (*i.e.* segmentation) shows the similarity in natural and

TABLE III
VALIDATION SEGMENTATION RESULTS

Template Subject Number	Segmentation Error Self Validation (seconds)		Segmentation Error Cross Validation (seconds)	
	Natural	Synth.	Natural	Synth.
Subj. 1	0.346	0.412	0.363	0.447
Subj. 2	0.392	0.325	0.549	0.509
Subj. 3	0.295	0.498	0.451	0.493
Subj. 4	0.333	0.339	0.492	0.521
Subj. 5	0.202	0.422	0.449	0.445

TABLE IV
MOVEMENT IDENTIFICATION RECALL AND PRECISION

Template Subject Number	Identification Recall Self Validation		Identification Recall Cross Validation	
	Natural	Synth.	Natural	Synth.
Subj. 1	100%	100%	100%	100%
Subj. 2	100%	100%	100%	95.8%
Subj. 3	100%	100%	91.7%	100%
Subj. 4	100%	100%	95.8%	100%
Subj. 5	83.3%	100%	100%	100%
Template Subject Number	**Identification Precision Self Validation**		**Identification Precision Cross Validation**	
	Natural	Synth.	Natural	Synth.
Subj. 1	100%	100%	88.9%	100%
Subj. 2	100%	100%	96%	100%
Subj. 3	85.7%	100%	96%	92.3%
Subj. 4	100%	96%	95.8%	96%
Subj. 5	100%	100%	100%	96%

synthesized data. The figure also shows the similarity in the data streams of the natural and synthesized data.

Table 3 shows the segmentation error for each of the movements. The starting and ending samples of each movement are determined by DTW. The difference between samples determined by DTW and the annotations were calculated and averaged for each scenario. The templates trained by each subject were compared with their own data (*i.e.* self-validation) and with the data from the other subjects (*i.e.* cross-validation) for the natural and synthesized data.

Table 4 shows the results of the movement identification from DTW. The recall and precision are shown in this table. Recall is defined as the number of true positives divided by the total number of positive instances (*i.e.* $TP/(TP + FN)$). Precision is defined as the number of true positives divided by the total number of elements selected (*i.e.* $TP/(TP + FP)$). For the trial sequences, the recall of detection was consistently high. Only the subject 5 based templates failed to identify all of the same subjects' movements. All but one of the movements that were not identified (*i.e.* false negatives) were the kneeling movement. This is true for both natural and synchronized data. The kneeling movement has the most variation due primarily to the depth differences for the subjects. Some kneeled with their knee touching the floor and others just enough to reach the floor with their hand. The precision was also high with very few false positives detected. Again, there is very little difference between the synthesized and natural data based on this measurement.

TABLE V
TRAINING DATA ACTIVITY RECOGNITION RESULTS

	Task Specific Template	MoST Template Time
Instances Found	23	24
Instances Missed	2	1
Detection Errors	1	1
Time Difference		
Minimum	**Mean**	**Maximum**
0 seconds	1.39 seconds	24 seconds

B. Training Data Validation

Beyond synthesis, the MoST dataset can be used directly as training data for algorithm development. To validate this use case, we created a stand-to-sit template using the MoST dataset. Another researcher that is recording long term data created training data (*i.e.* 10 repetitions) for the stand-to-sit movement. This data was collected at 40Hz with an accelerometer range of $\pm 4g$. This training data was also recorded in an environment with stools that are taller than the chairs used during the MoST data collection with the sensor in the subject's pocket. We use the x-axis of the accelerometer for both templates.

Using the right thigh sensor from the MoST dataset for the template, the MoST data is decimated and scaled to match the characteristics of the data in the in the system under test. The MoST based template and the template from the new training data are tested over a two days of data from the long term data collection. The continuous data from the first day had a single subject and lasted approximately four hours. The data from the second day had three subjects and had approximately eight hours and fifteen minutes of collection.

We use the DTW algorithm and thresholding to find instances of stand-to-sit in the long term data using the template from MoST and the template from the training data. We expect the template generated from MoST to perform similarly to the template generated specifically for this dataset. From the two days of data, there were 25 stand-to-sit instances observed on the video recording of the lab area used for the long term data collection. Using the MoST template, 24 of these instances were detected. Using the task specific templates, 23 of the instances were detected.

Table 5 shows the instance detection and error as well as the difference in time for the start of the instances found using both templates. The instances of the movement are expected to be found at the same start time if the templates behave the same. In the majority of cases, there is no difference in the time of the instances found. There is one outlier with a 24 second difference for the instances between the two templates.

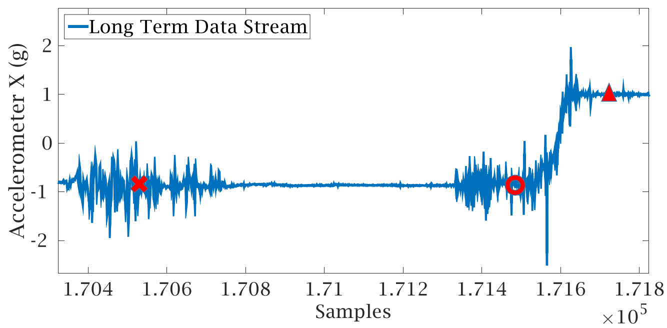


Fig. 10. Stand-to-Sit start times for outlier. The x marks the start based on the MoST template and the circle marks the start based on the data specific template. The triangle marks the end of the movement for both templates.

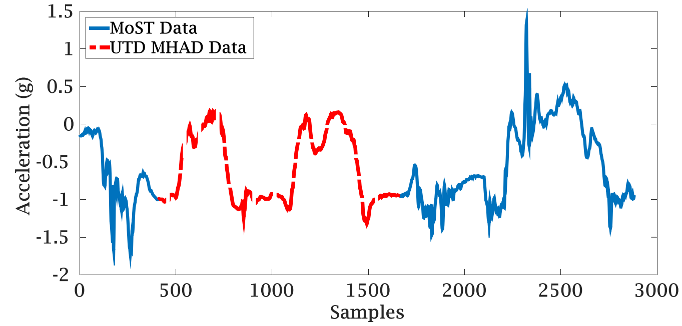


Fig. 11. Synthesis of wrist sensor data from MoST and UTD MHAD.

Fig. 10 shows the start times of this instance based on the MoST template (marked with x) and the data specific template (marked with a circle). This error is due to the MoST template matching a part of the signal that was before the actual stand-to-sit as well as the stand-to-sit. The subject stopped walking and stood for several seconds before sitting. The algorithm mapped the end of the walking data to the first samples of the MoST template. Because the rest of the movement before the stand-to-sit was relatively flat due to the subject standing still, the template was mapped to this segment and the DTW cost stayed below the threshold for this entire segment. This caused the difference in start time though the proper instance of the movement was detected. One other instance has a time difference of 2 seconds, and all other instances have a difference of 1 second or less. The average time difference overall instances is 1.39 seconds with a median value of 0 seconds. The template generated from MoST data would allow this algorithm to be tested without generating new templates and therefore save time for algorithm development.

A single application cannot prove the value of the data generated by MoST. However, we chose DTW for both validation scenarios as it is a robust algorithm for processing time-series and has been well studied. The use of this algorithm provides an example of how data synthesized from MoST can be used and at the same time shows the similarity between natural data and synthesized data. The results show that the MoST synthesis data is generally similar to the human data and can be useful for implementing and testing algorithms. It also shows that the MoST data can be used directly to save time and effort for recording training data for algorithm development even when there are differences in the sensor setup (*i.e.* range, sensitivity, frequency, location).

VI. Enhancements

MoST allows the synthesis of data sequences based on real sensor data that can be used for development of algorithms for wearable sensors. The MoST data can also be used to allow transfer learning [34]. The information gathered from this data can be adapted to and used in other training scenarios. Any algorithms developed or information gained from MoST data can be used to inform new algorithms. The development team as well as the open source community will contribute to the increased capabilities and quality of the system.

The primary enhancement is to increase the amount of data in the database. The increase in data will add further variability to the dataset. We are currently adding our own data, but we must ensure data from other sources is compatible. To test the ability of adding new data for use with MoST, we added the inertial sensor data from the University of Texas at Dallas Multimodal Human Action Dataset (UTD MHAD) [35]. This dataset has inertial data from a single sensor as well as Microsoft Kinect depth and RGB camera data. The dataset has 3-axis gyroscope and 3-axis accelerometer data collected with ranges of $\pm 1000\ deg/s$ and $\pm 8g$ respectively at 50Hz stored as MATLAB files. Updates to the Data Synthesis tool were made to handle synthesis using both datasets, which requires adjustments for range, sampling rate, and file type.

Fig. 11 shows an output of the Data Synthesis tool created using movements from the MoST dataset (blue, solid sections) as well as the UTD MHAD dataset (red, dashed section). This plot represents acceleration data from the x-axis of a sensor worn on the right wrist and shows how data from similar sensor locations in different datasets can be merged using the MoST tools. The sequence of movements is sit-to-stand, swipe right hand left (MHAD), swipe right hand right (MHAD), stand-to-sit, sit-to-lie, and lie-to-sit. Of the 27 movements in the UTD MHAD dataset, 23 of the movements differ from those already in MoST.

In the future, the toolset will accommodate collaborative efforts. Other researchers can add additional movements. As long as there is some overlap in sensor locations, there is no limit to the amount or type of movement and activities added. Additionally, the synthesis tool can be updated to output a variety of data rates and different bit depths to emulate varying sensor platforms. We are defining a file format that will more easily allow the community to add new data. The file will detail metadata about the subject (*e.g.*, age, sex, etc.). Additionally, the sensitivity and range of the sensors, sensor modalities and axes, and the orientation (relative to neutral posture) of the sensors should be defined in the file. While additional data can be added as discussed with manual updates, this updated file format and information will allow a new software tool to handle the conversion from the original data format into a MoST compatible format automatically. The Data Visualization tool must also be updated to handle videos that do not have frame timing information and data that does not have video.

With data coming from many sources, there is a risk of low quality data being added to the database. We will develop a confidence metric (*i.e.* ranking system) that allows the community to determine the quality of data in the system. The highest quality data will be selected for use in synthesis and lower quality data to be used less frequently. The community will validate the quality of the data through this system. Additionally, the community will be able to label and annotate data based on new needs and movement types.

Another difficulty of adding additional data is ensuring users are able to find the data they need in the system. Making improvements to the database search tool for sorting through large amounts of data will be critical. Additionally, the data used by the other tools in MoST will be stored in the HDF5 file format for simpler access. This hierarchical file format will allow for the structure of the data to be more easily understood without having to know about all of the data in the file. It also allows storage of the raw data, the meta-data, and the annotations in a single file.

Because some of our tools are built in MATLAB, we will also provide the executables for these functions to allow users without a license to have access to the related functionality. Considerations will be made in the future towards moving to license-free tools to allow editing by a larger group of users.

VII. Conclusions

The MotionSynthesis Toolset is an open source tool that should allow researchers and other users to synthesize large datasets with much less effort than standard data collection. They will also be able to contribute data that can be used in synthesis. The open source nature of the tool will allow for greater collaboration as well as improvement of the quality and amount of data in the database. Algorithm development and analysis can be hastened with the data from the tool.

MoST is available at http://motionsynthesis.org and on Github. The tool will be supported through additional data collection as well as efficiency and usability improvements for the tools. Community usage and feedback will add to the overall quality of the data and the usability of the toolset.

Acknowledgment

Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations.

References

[1] Y. Wang, L. Li, B. Wang, and L. Wang, "A body sensor network platform for in-home health monitoring application," in *Proc. 4th Int. Conf. IEEE Ubiquitous Inf. Technol. Appl. (ICUT)*, Dec. 2009, pp. 1–5.

[2] J. Wannenburg and R. Malekian, "Body sensor network for mobile health monitoring, a diagnosis and anticipating system," *IEEE Sensors J.*, vol. 15, no. 12, pp. 6839–6852, Dec. 2015.

[3] M.-M. Bidmeshki and R. Jafari, "Low power programmable architecture for periodic activity monitoring," in *Proc. ACM/IEEE 4th Int. Conf. Cyber-Phys. Syst.*, Apr. 2013, pp. 81–88.

[4] J. Mann, R. Rabinovich, A. Bates, S. Giavedoni, W. MacNee, and D. K. Arvind, "Simultaneous activity and respiratory monitoring using an accelerometer," in *Proc. Int. Conf. IEEE Body Sensor Netw. (BSN)*, May 2011, pp. 139–143.

[5] G. Bordello, W. Brunette, J. Lester, P. Powledge, and A. Rea, "An ecosystem of platforms to support sensors for personal fitness," in *Proc. Int. Workshop IEEE Wearable Implant. Body Sensor Netw. (BSN)*, Apr. 2006, pp. 174–178.

[6] T. R. Bennett *et al.*, "MotionSynthesis toolset (MoST): A toolset for human motion data synthesis and validation," in *Proc. 4th ACM MobiHoc Workshop Pervasive Wireless Healthcare (MobileHealth)*, New York, NY, USA, 2014, pp. 25–30. [Online]. Available: http://doi.acm.org/10.1145/2633651.2637472

[7] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, Aug. 2004, pp. 32–36.

[8] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Oct. 2005, pp. 1395–1402.

[9] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-51949084792&partnerID=40&md5=22e680a9d509bcd8686307cc24d702ad

[10] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[11] O. Kliper-Gross, T. Hassner, and L. Wolf, "The action similarity labeling challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 615–621, Mar. 2012.

[12] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 2556–2563.

[13] P. Kuryloski *et al.*, "DexterNet: An open platform for heterogeneous body sensor networks and its applications," in *Proc. 6th Int. Workshop IEEE Wearable Implant. Body Sensor Netw. (BSN)*, Jun. 2009, pp. 92–97.

[14] A. Y. Yang, S. Iyengar, P. Kuryloski, and R. Jafari, "Distributed segmentation and classification of human actions using a wearable motion sensor network," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2008, pp. 1–8.

[15] W. Ugulino, D. Cardador, K. Vega, E. Velloso, R. Milidiú, and H. Fuks, "Wearable computing: Accelerometers' data classification of body postures and movements," in *Advances in Artificial Intelligence—SBIA*. Berlin, Germany: Springer-Verlag, 2012, pp. 52–61.

[16] M. Zhang and A. A. Sawchuk, "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proc. ACM Conf. Ubiquitous Comput.*, 2012, pp. 1036–1043.

[17] M. Atzori *et al.*, "Characterization of a benchmark database for myoelectric movement classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 1, pp. 73–83, Jan. 2015.

[18] T. V. Marcard, G. Pons-Moll, and B. Rosenhahn, "Multimodal motion capture dataset TNT15," Leibniz Univ. Hannover, Hanover, Germany, and Max Planck for Intelligent Systems, Tübingen, Germany, Tech. Rep., 2016.

[19] P. Zappi, C. Lombriser, E. Farella, L. Benini, and G. Tröster, "Experiences with experiments in ambient intelligence environments," in *Proc. IADIS Int. Conf. Wireless Appl. Comput.*, 2009, pp. 171–174.

[20] S. S. Intille *et al.*, "Using a live-in laboratory for ubiquitous computing research," in *Pervasive Computing*. Berlin, Germany: Springer-Verlag, May 2006, pp. 349–365.

[21] F. De la Torre Frade *et al.*, "Guide to the Carnegie Mellon University multimodal activity (CMU-MMAC) database," Robot. Inst., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-08-22, 2008.

[22] D. Roggen *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *Proc. 7th Int. Conf. IEEE Netw. Sens. Syst. (INSS)*, Jun. 2010, pp. 233–240.

[23] D. Takada, T. Ogawa, K. Kiyokawa, and H. Takemura, "A context-aware AR navigation system using wearable sensors," in *Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction*. Springer, 2009, pp. 793–801.

[24] C. Doukas and I. Maglogiannis, "Managing wearable sensor data through cloud computing," in *Proc. IEEE 3rd Int. Conf. IEEE Cloud Comput. Technol. Sci. (CloudCom)*, Nov./Dec. 2011, pp. 440–445.

[25] V. Loseu, H. Ghasemzadeh, and R. Jafari, "A mining technique using N n-grams and motion transcripts for body sensor network data repository," *Proc. IEEE*, vol. 100, no. 1, pp. 107–121, Jan. 2012.

[26] V. Loseu, J. Mannil, and R. Jafari, "Lightweight power aware and scalable movement monitoring for wearable computers: A mining and recognition technique at the fingertip of sensors," in *Proc. 2nd Conf. Wireless Health. ACM*, 2011, p. 7.

[27] M. L. Sbodio and W. Thronicke, "Ontology-based context management components for service oriented architectures on wearable devices," in *Proc. 3rd IEEE Int. Conf. Ind. Informat. (INDIN)*, Aug. 2005, pp. 129–133.

[28] K. Murao, Y. Takegawa, T. Terada, and S. Nishio, "CLAD: A sensor management device for wearable computing," in *Proc. 7th Int. Workshop Smart Appl. Wearable Comput. (IWSAWC)*, Jun. 2007, p. 46.

[29] D. Bannach, "Tools and methods to support opportunistic human activity recognition," Ph.D. dissertation, Dept. Fachbereich Informatik, Kaiserslautern Univ. Technol., Kaiserslautern, Germany, 2015.

[30] D. Foti and L. Kanazawa, "Activities of daily living," in *Pedretti's Occupational Therapy: Practice Skills for Physical Dysfunction*, 6th ed. 2008, pp. 146–194.

[31] M. Folk, G. Heber, Q. Koziol, E. Pourmal, and D. Robinson, "An overview of the HDF5 technology suite and its applications," in *Proc. EDBT/ICDT Workshop Array Databases*, 2011, pp. 36–47.

[32] C. Berge and E. Minieka, *Graphs and Hypergraphs*, vol. 7. Amsterdam, The Netherlands: North Holland, 1973.

[33] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proc. KDD Workshop*, 1994, vol. 10. no. 16, pp. 359–370.

[34] L. Zhang, W. Zuo, and D. Zhang, "LSDT: Latent sparse domain transfer learning for visual adaptation," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1177–1191, Mar. 2016.

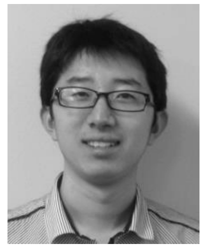
[35] C. Chen, R. Jafari, and N. Kehtarnavaz, "UTD-MHAD: A multi-modal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 168–172.

Terrell R. Bennett (S'14) received the S.B. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, in 2002, and the M.S. degree in electrical engineering from the University of Texas at Dallas, in 2007.

He was an Electrical Design Engineer with the industry from 2007 to 2013. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering, University of Texas at Dallas. His research is focused on signal processing and algorithm design for synchronization of cyber-physical systems and the Internet of Things as well as working with data from wearable sensors to detect activities, estimate motion, and improve the quality of the sensor data.

Mr. Bennett received the Texas Analog Center of Excellence Analog Fellowship, the Best of Doctoral Colloquium Award at Body Sensor Networks 2014, and the Best Presentation of Session Award at the American Control Conference 2013.

Hunter C. Massey (S'15) received the bachelor's degree in computer engineering from the University of Texas at Dallas, Richardson, TX, in 2014, where he is currently pursuing the M.S. degree in computer engineering. His research interests lie in signal processing and data analysis.



Jian Wu received the M.S. degree in communication and information systems from the Huazhong University of Science and Technology, Wuhan, China, in 2012. He is currently pursuing the Ph.D. degree in computer engineering with Texas A&M University. His research interests include signal processing algorithm development for IMU-based movement monitoring and activity recognition.

Syed Ali Hasnain (S'16) received the bachelor's degree in electrical engineering from the National University of Science and Technology, Pakistan. He is currently pursuing the Ph.D. degree in computer engineering with Texas A&M University. His research interests include wearable computing, context awareness, embedded systems, and signal processing.

Roozbeh Jafari (SM'12) received the Ph.D. degree in computer science from UCLA. He completed a post-doctoral fellowship at UC-Berkeley. He is an Associate Professor of Biomedical Engineering, Computer Science and Engineering, and Electrical and Computer Engineering with Texas A&M University. His research interest lies in the areas of wearable computer design and signal processing. His research has been funded by the NSF, NIH, DoD (TATRC), AFRL, AFOSR, DARPA, SRC, and industry (Texas Instruments, Tektronix, and Samsung & Telecom Italia). He has authored over 100 papers in refereed journals and conferences. He served as the General Chair and Technical Program Committee Chair for several flagship conferences in the area of wearable computers, including the ACM Wireless Health 2012 and 2013, the International Conference on Body Sensor Networks 2011, and the International Conference on Body Area Networks 2011. He was a recipient of the NSF CAREER Award in 2012, the IEEE Real-Time and Embedded Technology and Applications Symposium Best Paper Award in 2011, and the Andrew P. Sage Best Transactions Paper Award from the IEEE Systems, Man and Cybernetics Society in 2014. He is an Associate Editor of the IEEE SENSORS JOURNAL, the IEEE INTERNET OF THINGS JOURNAL, and the IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS.