Robust Activity Recognition using Wearable IMU Sensors

Yashaswini Prathivadi, Jian Wu, Terrell R. Bennett, Roozbeh Jafari
Department of Electrical Engineering
University of Texas at Dallas
Dallas, USA
{yxp120330, jian.wu, tbennett, rjafari}@utdallas.edu

***Abstract*— In this paper, an orientation transformation (OT) algorithm is presented that increases the effectiveness of performing activity recognition using body sensor networks (BSNs). One of the main limitations of current recognition systems is the requirement of maintaining a known, or *original*, orientation of the sensor on the body. The proposed OT algorithm overcomes this limitation by transforming the sensor data into the *original* orientation framework such that orientation dependent recognition algorithms can still be used to perform activity recognition irrespective of sensor orientation on body. The approach is tested on an orientation dependent activity recognition system which is based on dynamic time warping (DTW). The DTW algorithm is used to detect the activities after the data is transformed by OT. The precision and recall for the activity recognition for five subjects and five movements was observed to range from 74% to 100% and from 83% to 100%, respectively. The correlation coefficient between the transformed data and the data from the *original* orientation is above 0.94 on axis with well-defined patterns.**

Keywords—Activity recognition, IMU sensors, Orientation transformation

I. Introduction

Activity recognition is an important application providing interesting ways to assess health and wellness of an individual. It has monumental impact on longitudinal studies for those neurodegenerative disorders that affect motor functionality. Wearable sensors with inertial measurement units (IMUs) consisting of 3-axis accelerometers and 3-axis gyroscopes provide an unobtrusive way to monitor and detect activities.

There are several sensor orientation dependent activity recognition systems that use data from sensors to perform activity recognition [1, 2, 3]. These techniques perform accurate activity recognition when the sensor orientation is maintained according to the orientation used in the training phase. However, if the sensor orientation on the body is changed due to user's movements or because of incorrect sensor placement by the user, these activity recognition techniques can fail. This drawback is more significant in the case where data is collected to perform longitudinal study because such valuable data cannot be generated again. Hence an orientation independent way of performing activity recognition is needed.

Throughout this paper, the *original* orientation describes the placement of the sensor during initial set up and training phase of the algorithm. The orientation dependent activity recognition algorithms are trained to recognize activities in the *original* orientation. The *new* orientation describes the placement of the sensor after misplacement on the same body part. The algorithms trained by the *original* orientation will have to be applied to the readings acquired in the *new* orientation. This discrepancy between the *original* and the *new* orientation introduces challenges for activity recognition.

The main contribution of this paper is the orientation transformation (OT) algorithm. In our approach, we learn the orientation difference between the *original* orientation and the *new* orientation of the sensor when the user performs a known movement in the *new* orientation. This difference is quantified as a rotation matrix (R-matrix). The R-matrix is used to transform the data collected in the *new* orientation into the *original* orientation. This ensures that the orientation dependent activity recognition techniques are now independent of sensor orientation changes, making it more robust. The user could wear the sensor with any orientation. Our technique requires that the sensor is rigidly attached to the body. It cannot handle scenarios where the sensor is not firmly attached or it is loose or wobbling.

There are two main limitations of our approach. First is that it requires sensor readings for two known postures in the *new* orientation of the sensor. The second limitation is that the two postures should be such that the accelerometer readings in those postures have a non-zero angular rotation or difference. These limitations, however, are easily addressable. Once the user wears the sensor in any *new* orientation he/she can be asked to perform a certain movement like stand-to-sit such that the two postures *stand* and *sit* can be noted in the first few sensor readings.

The remainder of this paper is structured as follows. Related works are discussed in section II. Section III covers our proposed algorithm. In section IV, we describe the experimental setup and discuss the experimental results. Lastly, in section V, we present our conclusions and future work.

II. Related Works

In some prior research, statistical techniques are used to extract orientation independent features from the sensor data and train classifiers to perform activity recognition [4]. The feature set used is limited and better recognition results for more activities can be obtained by using more orientation dependent features [3]. Additionally, these techniques are standalone techniques and do not work in conjunction with other orientation dependent techniques already available.

There have been some other works that leverage transformation techniques added to the existing activity recognition algorithms. In some of these studies restrictions are placed on the number of orientations the sensor could have if misplaced. For each of these orientations the corresponding R-matrix is already calculated [5]. The gravity vector representation from the sensor is used to match one of the known orientations. The corresponding R-matrix is used to transform the data. Activity recognition is then performed by a classifier which is trained using data from each of the sensor orientations after appropriate data transformations. This setup restricts orientation changes on the sensor and hence cannot be applied to scenarios where there could be more flexibility in placing the sensor on the body.

Some researchers have also looked at achieving orientation independent activity recognition by detecting cyclic movements like walking and learning the R-matrix representing the difference between the *original* orientation and the *new* orientation using the statistics derived from walking data [6]. However an assumption is made that at any point in time, the orientation of the sensor which is placed on the thigh can change by rotating only about one coordinate axis. This assumption does not apply to those cases where the sensors are of such small form-factors that they could have rotations about multiple axes when placed on the body.

In comparison to these investigations, our approach has a few advantages. It acts as an add-on to existing orientation dependent systems and facilitates their use without imposing any constraints on the sensor orientation on the body. No prior knowledge about *new* orientation is needed. Furthermore, no assumptions on any specific axis of rotation for orientation change of sensors are made.

III. Proposed Approach

In our approach, we learn the orientation difference between the *new* and the *original* orientation and quantify it in an R-matrix. In order to learn the orientation difference between two Cartesian coordinate systems, a common inertial reference system is needed. When the sensor is placed on the body, there is no known common reference system between the *original* and *new* orientations. The movements could be treated as the common aspect between the *original* and the *new* orientation. Hence, movements that are found in both orientations are used to construct the shared reference system.

Let the *original* sensor orientation coordinate system be denoted as *S1* and the *new* sensor orientation coordinate system be denoted as *S2*. A representation of the inertial reference system and the sensor coordinate system in both the *original* orientation and the *new* orientation can be seen in Fig. 1. The *x, y* and *z* axes of the common 3D inertial reference system are represented as X^i, Y^i and Z^i. X^{S1}, Y^{S1} and Z^{S1} represent *x, y* and *z* axes of the *original* orientation, *S1*, of the sensor. X^{S2}, Y^{S2} and Z^{S2} represent *x, y* and *z* axes of the *new* orientation, *S2,* of the sensor.

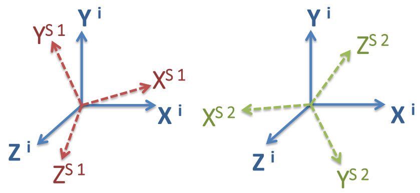

Fig. 1. Inertial reference system vs. sensor orientations *S1* and *S2*

Finding the orientation difference between the *new* and the *original* sensor orientation can be broken down into three steps that are described in detail below.

A. Define a common inertial reference frame

The common inertial frame of reference is constructed using common movements performed in both orientations. Let us consider one such movement that was done in both orientations, say a sit-to-stand movement. The information common to both orientations in this movement are: the ***initial posture*** (sitting, annotated as posture-1) and the ***final posture*** (standing, annotated as posture-2). The inertial frame of reference with respect to any sensor orientation is defined:

Definition 1*: Given two postures posture-1 and posture-2, and the normalized acceleration vectors in those postures, such that the angular difference between them is non-zero, the x, y, and z axes of the inertial frame of reference are given as:*

X Axis: Cross product of the normalized acceleration vectors for posture-1 and posture-2

Z Axis: Normalized acceleration vector for posture-1

Y Axis: Cross product of Z and X axis.

These acceleration vectors represent the effect of only gravity on the sensors in those postures. Consider the following example that illustrates the construction of the inertial reference system. The movement used is sit-to-stand: In sensor orientation *S* of the sensor, let a_1^s be the normalized acceleration vector for posture-1 (sitting posture) and let a_2^s be the normalized acceleration vector for the posture-2 (standing posture). Let X_i^s, Y_i^s and Z_i^s be the representation of the *x*, *y*, and *z* axes of the inertial reference frame with respect to the orientation of the sensor (i.e. *S*). As shown in Fig. 2, using Definition 1, the inertial frame of reference is constructed as

$$X_i^s = a_1^s \times a_2^s \tag{1}$$

$$Z_i^s = a_1^s \tag{2}$$

$$Y_i^s = Z_i^s \times X_i^s \tag{3}$$

where $\times$ indicates vector cross product and a_1^s and a_2^s are the row vectors

$$a_1^s = [a_{1x}^s \quad a_{1y}^s \quad a_{1z}^s] \tag{4}$$

$$a_2^s = [a_{2x}^s \quad a_{2y}^s \quad a_{2z}^s] \tag{5}$$

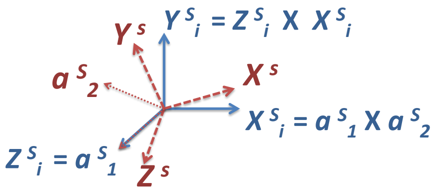

Fig. 2. Inertial reference frame in-terms of the orientation axes system *S*

This process is done for the *original* and the *new* sensor orientations (*i.e. S1* and *S2*). Thus the inertial frame of reference that is common to both the *original* orientation and the *new* orientation is constructed in both orientations as X_i^{s1}, Y_i^{s1}, Z_i^{s1}, X_i^{s2}, Y_i^{s2}, and Z_i^{s2} and all are 3×1 row vectors.

B. Derive R-matrix for the orientation difference between each of the sensor orientations and the inertial reference frame

The orientation difference is calculated using the constructed inertial frame of reference.

Let *R* be the rotation matrix describing the rotation from *orientation S* to the inertial reference frame described as

$$R = \begin{bmatrix} X^i \cdot X^S & X^i \cdot Y^S & X^i \cdot Z^S \\ Y^i \cdot X^S & Y^i \cdot Y^S & Y^i \cdot Z^S \\ Z^i \cdot X^S & Z^i \cdot Y^S & Z^i \cdot Z^S \end{bmatrix} \tag{6}$$

R can also be written in terms of X_i^s, Y_i^s and Z_i^s calculated in (1) ,(2) and (3) as

$$R = \begin{bmatrix} X_i^s \\ Y_i^s \\ Z_i^s \end{bmatrix} \tag{7}$$

This process is completed for the *original* and the *new* sensor orientations (*i.e. S1* and *S2*) to get corresponding rotation matrices $R_1$and R_2.

*C. Find the R-matrix representing the orientation difference between the **new** and the **original** orientation.*

The representation of the rotation matrix *R* for the orientation difference between *S2* and *S1* is given as

$$R = \begin{bmatrix} X^{S1} \cdot X^{S2} & X^{S1} \cdot Y^{S2} & X^{S1} \cdot Z^{S2} \\ Y^{S1} \cdot X^{S2} & Y^{S1} \cdot Y^{S2} & Y^{S1} \cdot Z^{S2} \\ Z^{S1} \cdot X^{S2} & Z^{S1} \cdot Y^{S2} & Z^{iS1} \cdot Z^{S2} \end{bmatrix} \tag{8}$$

This can also be calculated using *R1* and *R2* as

$$R = R_1' * R_2 \tag{9}$$

where R_1' represents transpose of the matrix R_1. The R-matrix in (9) is multiplied with acceleration vectors collected in *S2* (*new* orientation), to transform readings to orientation system *S1* (*original* orientation).

IV. Experiments and Results

We conducted experiments to validate the OT algorithm. In these experiments, data collection was performed using IMU sensors with a 3-axis accelerometer and a 3-axis gyroscope at a sampling rate of 20Hz.

The first set of experiments was designed to test the data transformation of the OT algorithm. Two sensors, *S1* and *S2*, were glued to each other in different orientations and attached to the right thigh of a subject. The subject was then asked to perform the stand-to-sit and sit-to-stand movements ten times. Since the orientations were different, the same movements had different patterns from the view of each sensor. The OT transformation algorithm was applied, and the orientation difference between the two sensor orientations was learned in the form of the R-matrix. The R-matrix was used to transform the sensor data collected in sensor *S2* orientation to the sensor *S1* orientation. The correlation coefficient for the signals on each of the axis is presented in Table I.

TABLE I. Correlation Coefficient Value Between Data From Sensor *S1* And The Transformed Data From *S2*

Axis	Correlation Coefficient
Accelerometer-XAxis	0.9807
Accelerometer-YAxis	0.4941
Accelerometer-ZAxis	0.9968
Gyroscope-XAxis	0.7939
Gyroscope-YAxis	0.9442
Gyroscope-ZAxis	0.5930

From Table I, it can be seen that in the accelerometer's X and Z axes and the gyroscope's Y axis, the correlation coefficients are higher than on the other axes since the major component of the movement occurred on these axes. On the remaining axes, the movement performed does not create well defined repeatable patterns resulting in the lower correlation coefficients. These similarity measures show that the OT algorithm accurately transforms the data.

We also conducted experiments to check the effectiveness of the algorithm in empowering orientation dependent activity recognition algorithms. The algorithm we chose for this purpose is based on DTW [7, 8, 9]. We used the SPRING algorithm to implement DTW for these experiments [10]. Data collection for this experiment was performed as follows. Five subjects were asked to wear three sensors, one on the right thigh, one on the waist, and the last on the right ankle. Three sessions of data collection were performed on each subject. The movements performed in each session were stand-to-sit, sit-to-stand, sit-to-lie, lie-to-sit, and kneel.

At the beginning of each session, the orientations of the sensors were changed to a random state. Each movement was performed ten times in each session. Per subject, 30 total instances of each movement were collected from the sessions. Thus from all the subjects, a total of 150 instances of each movement were collected from all of the sessions. In order to perform activity recognition for each subject, the template for each type of movement was learned from the first session. Data from the second and third sessions were transformed into the orientation from the first session using the OT algorithm. The learned templates were applied to data from first session and the transformed data from the other sessions to perform activity recognition. The precision (Pr) and recall (Rc) for the activity recognition performed for each subject (Sub1-5) for each movement (M) in the order given above are shown in Table II.

TABLE II. SUBJECT-WISE ACTIVITY RECOGNITION RESULTS

	Sub1		Sub2		Sub3		Sub4		Sub5	
M	Pr	Rc	Pr	Rc	Pr	Rc	Pr	Rc	Pr	Rc
1	100	97	100	100	100	100	100	100	100	100
2	100	100	100	100	100	91	100	91	100	97
3	100	100	100	100	100	100	100	100	74	83
4	100	100	100	100	100	100	77	100	91	97
5	100	100	100	100	100	100	100	100	100	100

From Table II, we can see that DTW yields good precision and recall with the OT algorithm. The DTW-distance histogram is used as a metric to evaluate the performance of DTW. Better separation and discrimination between the target movements and non-target movements is obtained with a larger DTW-distance. The histogram of DTW-distance obtained while searching for stand-to-sit movement from one of the data collection sessions is shown in Fig. 3. The OT algorithm was used to transform the data prior to the calculation of the DTW-distance. The white bars in Fig. 3 highlight the DTW-distance bins (with respect to the template) for the target-movement instances in the transformed movement data. The black bars indicate the DTW-distance bins for non-target movement samples. The separation between the black and the white bars are an indication of the performance of the DTW. A good separation tells us that the OT algorithm successfully transforms the movement data to the *original* orientation and that the target movements can be successfully distinguished.

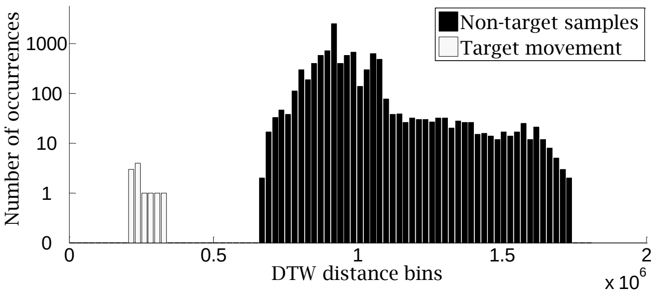

Fig. 3. DTW distance histograms for the transformed data

For comparison, a histogram was plotted for the DTW-distance obtained while searching for the stand-to-sit target-movement in movement data which was collected in the *original* orientation of the sensor. Here, no transformation technique was performed. This histogram is shown in Fig. 4. From Fig. 3 and Fig. 4, we can see that the OT algorithm provides good separation between the histogram bins and that the DTW performance is not affected by the OT algorithm.

V. CONCLUSION AND FUTURE WORK

In this paper we presented the orientation transformation (OT) algorithm to transform sensor data from a *new* orientation into the *original* orientation. Our experimental results show that our algorithm successfully transforms the sensor data and can enable an orientation dependent activity recognition algorithm like DTW to achieve suitable recognition accuracy with orientation changes in the sensor. However, this algorithm makes some assumptions on the availability of posture information. While this can be easily addressed by enforcing a calibration movement, it creates an inconvenience for users. For seamless online activity recognition without any user intervention, orientation independent posture recognition has to be incorporated. Our future work comprises of finding a posture recognition technique to make the OT algorithm even more robust.

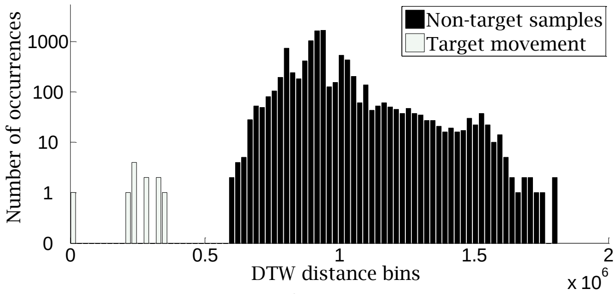

Fig. 4. DTW distance histograms for movement data in *original* orientation

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation, under grants CNS-1150079 and CNS-1012975, and the TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. 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] E. Guenterberg, H. Ghasemzadeh, and R. Jafari, “A distributed hidden markov model for fine-grained annotation in body sensor networks,” in *Wearable and Implantable Body Sensor Networks, 2009. BSN 2009. Sixth International Workshop on*, pp. 339–344, IEEE, 2009.

[2] R. Lotfian and R. Jafari, “An ultra-low power hardware accelerator architecture for wearable computers using dynamic time warping,” in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 913–916, EDA Consortium, 2013.

[3] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, “A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer,” *Information Technology in Biomedicine, IEEE Transactions on*, vol. 14, no. 5, pp. 1166–1172, 2010.

[4] J. Yang, “Toward physical activity diary: motion recognition using simple acceleration features with mobile phones,” in *Proceedings of the 1st international workshop on Interactive multimedia for consumer electronics*, pp. 1–10, ACM, 2009.

[5] S. Thiemjarus, “A device-orientation independent method for activity recognition,” in *Body Sensor Networks (BSN), 2010 International Conference on*, pp. 19–23, IEEE, 2010.

[6] X. Wu, Y. Wang, C. Chien, and G. Pottie, “Self-calibration of sensor misplacement based on motion signatures,” in *Body Sensor Networks (BSN), 2013 IEEE International Conference on*, pp. 1–5, IEEE, 2013.

[7] D. J. Berndt and J. Clifford, “Using dynamic time warping to find patterns in time series.,” in *KDD workshop*, vol. 10, pp. 359–370, Seattle, WA, 1994.

[8] E. Keogh and C. A. Ratanamahatana, “Exact indexing of dynamic time warping,” *Knowledge and information systems*, vol. 7, no. 3, pp. 358–386, 2005.

[9] S. Salvador and P. Chan, “Toward accurate dynamic time warping in linear time and space,” *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

[10] Y. Sakurai, C. Faloutsos, and M. Yamamuro, “Stream monitoring under the time warping distance,” in *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pp. 1046–1055, IEEE, 2007.