Position Invariance for Wearables: Interchangeability and Single-Unit Usage via Machine Learning

Aras Yurtman⬤, Billur Barshan⬤, and Soydan Redif⬤, *Senior Member, IEEE*

Abstract—We propose a new methodology to attain invariance to the positioning of body-worn motion-sensor units for recognizing everyday and sports activities. We first consider random interchangeability of the sensor units so that the user does not need to distinguish between them before wearing. To this end, we propose to use the compact singular value decomposition (SVD) that significantly reduces the accuracy degradation caused by random interchanging of the units. Second, we employ three variants of a generalized classifier that requires wearing only a single sensor unit on any one of the body parts to classify the activities. We combine both approaches with our previously developed methods to achieve invariance to both position and orientation, which ultimately allows the user significant flexibility in sensor-unit placement (position and orientation). We assess the performance of our proposed approach on a publicly available activity data set recorded by body-worn motion-sensor units. The experimental results suggest that there is a tolerable reduction in accuracy, which is justified by the significant flexibility and convenience offered to users when placing the units.

Index Terms—Accelerometer, activity monitoring and classi-fication, gyroscope, inertial sensors, Internet of Things (IoT), machine learning classifiers, magnetometer, motion sensors, ori-entation invariance, pattern recognition, position invariance, wearable sensing.

I. Introduction

WITH the growth and proliferation of devices in homes, factories, hospitals, vehicles, etc., there is an ever increasing need for solutions to device/data integration and management. Wearables containing devices such as motion sensors are considered a key element of Internet of Things (IoT), which play a significant role in ambient intelligence and assistive technology [1], [2]. Pervasiveness of communi-cating sensors and computing intelligence has facilitated the integration of motion sensors in devices or accessories used in everyday life. Through a network of intelligent sensors, valu-able information about the user state and well being can be obtained.

Designing context-aware systems that can monitor, under-stand, and classify human activities correctly is essential to improve the user's quality of life. Monitoring daily living activities and detecting unexpected behavior or circumstances to support the elderly population or those with special needs are challenging research issues [3], [4]. Personal safety and comfort of the user must not be compromised by limiting their independence, restricting their mobility, invading their privacy, and/or degrading their quality of life.

Superior in several respects to complementary sensors-based smart environments [5], recognizing human activity using wearable motion sensors has recently received growing atten-tion. Wearable systems offer a more feasible solution, both in terms of cost and infrastructure [6]. Body-worn sensors have become smaller, lighter, less costly, wireless, and portable with reduced power consumption. The concerns related to dis-comfort, portability, and affordability are lessened with such improvements. Wearables can be readily integrated with pop-ular user devices and accessories, such as mobile phones and smart watches, as explored in, e.g., [7] and [8]. In [8], for example, an IoT-based wireless system (SmartPANTS) is developed as a smart home-rehabilitation platform for use in the physical therapy of recovering stroke patients.

A. Main Focus

Overall, wearing sensors is a suitable option for auto-mated monitoring and recognition of activities, as advocated in our earlier works [9]–[11] and championed in this study. However, a pertinent issue, related to the focus of this article, is the proper attachment of the wearables by users. Wearable systems often require the user to affix multiple sensor units to predetermined positions and orientations on their body. However, sensor-unit affixment deviates from the ideal config-uration from user to user, which can degrade wearable system performance. Expecting the user to identify and place multiple units correctly each time at predetermined positions and ori-entations can be restrictive, tedious, and overwhelming. This is especially true for elderly, injured, or disabled users in daily activity monitoring and fall detection [12], [13] or for telerehabilitation [14].

Even with correctly affixed sensor units, their placement (position and orientation) may inevitably change with time

due to movement, vibrations, impacts on the body, etc. If the devices are not directly worn on the skin but affixed to the user's clothing or accessories, the problem becomes more severe. Sensor-unit placement poses a significant issue involving smartphones due to the obvious variability in position and orientation of the phone [15]–[17]. In [17], a smartphone-based activity recognition method is described where there are a limited number of allowable positions/orientations to distinguish four daily activities. Note that the consideration of only a restricted set of predetermined positions and/or orientations is a trait found in most previous studies.

Hence, it is clear that when affixing sensors, the need to provide positional and/or orientational flexibility to users of wearable-based activity recognition systems arises [18]. Furthermore, this would reduce the demand on the amount of training data/time since the wearable system would not need to be trained for every possible sensor-unit placement.

B. Related Work

Activity recognition methods that are robust to errors due to sensor-unit placement uncertainty have received limited attention [19]. Our recent works have addressed the problem of invariance to position [20] and to orientation [21], [22] of body-worn motion-sensor units, ultimately providing users with a degree of pliability and freedom.

A detailed survey on methods for attaining invariance to the positioning of body-worn motion-sensor units is provided in [20] and [24]. One type of method gains invariance to sensor-unit positioning by transforming the sensory data and extracting heuristic features based on the transformed data [25], [26].

Alternatively, one could train classifiers to distinguish between different sensor-unit positions. One way of accomplishing this is to use a single *generalized classifier* to aggregate all sensor-data features for all possible sensor-unit positions.

A number of works have also utilized differing sensor-placement configurations at different times and/or places, and/or from different subjects [27]–[30].

Chavarriaga *et al.* [31] and Förster *et al.* [32] claim that class means are shifted in the feature space as a consequence of position variations. Acceptable results are obtained for position shifts; however, unsatisfactory classification is observed when the sensor units are moved across different body parts, which is expected.

Another group of studies is based on the assumption that *a priori* information about the sensor units and their positions is available, which is not realistic [33]. Invariance to sensor-unit orientation is achieved via the use of the magnitude of the acceleration vector [34]. Multistage processing has also been employed, where a stage is used for classifying the different sensor-unit positions and another stage used for activity recognition [35]–[37].

Earlier-developed methods focusing on position invariance are not comparable with each other due to a number of reasons, such as sensor type and placement configuration, classification, and cross-validation process. In addition, the effect of the proposed approach on the activity recognition rate is not always made clear. Another issue is that existing studies on position invariance make different assumptions about the unit orientation [32], [35], [38], [39].

Some studies consider a very limited number of activities [28] or consider only the simple ones, even though these may impact classification accuracy greatly [30]. Activities that are difficult to classify and often confused with others are commonly merged into fewer classes to bolster classification accuracy [29], [34], [40]. Conversely, in this article, we consider a data set comprising a broad range of static (stationary) as well as dynamic activities (see Section V-A). To simultaneously classify these types of activity is more difficult than those considered in studies to date. Unlike most existing works that rely on solely accelerometer data, we also use gyroscope as well as magnetometer data to gain invariance to sensor-unit placement in activity recognition.

C. Main Contributions and Structure

In this article, we develop techniques that improve the flexibility offered to the user in the positioning of wearable motion-sensor units on different rigid-body parts. Specifically, the salient features of this article are fourfold.

1) A technique for gaining position invariance, in terms of sensor-unit interchangeability on body parts. As such, the user need not keep track of the various unit locations.
2) A method for activity classification using only a single sensor unit without exploiting *a priori* information about which body part the unit is affixed. This is in line with the recent trend of using only a single sensor unit.
3) A methodology to accommodate shifting in sensor-unit positions while still on the same rigid body part, with ideal orientations maintained [20].
4) A technique for both position and orientation invariance on a broad range of static (stationary) and dynamic activities in our activity data set, made available in IEEE DataPort [23]. The simultaneous classification of such activities is a more challenging problem compared to those considered in most studies to date. The proposed method can provide invariance over a continuum of different positions and orientations.

These contributions constitute a generic framework for the provision of significant flexibility in the placement of sensor units on the body, which has been lacking in existing approaches till now.

The remainder of this article is organized as follows. Section II provides essential background material. The proposed algorithm, based on the compact singular value decomposition (SVD), is described in Section III. Section IV presents three variants of a generalized activity classifier that uses only a single sensor unit. The data set and our activity recognition methodology are detailed in Section V. The proposed schemes are evaluated in Sections VI and VII. Discussion and concluding remarks are provided in Section VIII.

II. BACKGROUND

Rigid body motion in 3-D can be decomposed into translational and rotational components [41]. Translations and

rotations are *rigid body transformations* during which the distance between any two arbitrary points on the rigid body is preserved. In our context, the relative position between any two arbitrary points on the *same* body part does not change with respect to time during motion. Accordingly, the head, torso, and lower/upper limbs would be considered to be rigid body parts.

Motion-sensor units produce outputs that are directly related to the linear and angular motion of the structure to which they are affixed. If the structure is rigid, all of the points comprising the rigid body have identical linear and identical angular velocity. The magnitude of the 3×1 angular velocity (rate) vector $\boldsymbol{\omega}$ corresponds to the rate of rotation and the instantaneous axis of rotation of the structure determines its direction. Accelerometers, gyroscopes, and magnetometers are commonly employed motion-sensor types, and are typically combined in a wearable sensor unit. A gyroscope with three mutually perpendicular axes directly provides the components of the $\boldsymbol{\omega}$ vector at its output.

A magnetometer detects the vectoral sum of the Earth's magnetic field vector $\mathbf{m}$ superimposed with external magnetic fields or disturbances (if any). The vector $\mathbf{m}$ points to the magnetic north and its magnitude and direction do not change significantly with the position of the sensor unit not only on a given rigid body part but throughout the human body. Thus, the three components of the vector $\mathbf{m}$, detected with reference to the sensor-unit frame, do not depend on the position of the magnetometer, but only on its orientation on the body.

Given that the angular velocity and magnetic field vectors and their magnitudes ($\boldsymbol{\omega}$, $\mathbf{m}$, $|\boldsymbol{\omega}|$, and $|\mathbf{m}|$, respectively, where $|\cdot|$ denotes the Euclidean norm of a vector) are not dependent on how the sensor unit is positioned on a given rigid body part, it is straightforward to use them directly to attain position invariance in activity recognition. Contrarily, the acceleration vector and its magnitude ($\mathbf{a}$ and $|\mathbf{a}|$, respectively) are dependent on position and the activity recognition rate deteriorates when they are directly used for the same purpose. In our earlier work on achieving position invariance for activity recognition [20], [24], we have extracted and used sequence combinations alternative to the raw acceleration vector $\mathbf{a}$ that are not influenced by the positioning of the sensor unit on a given rigid body part. More specifically, we have identified and used the projection of the total acceleration vector $\mathbf{a}$ in the direction of $\dot{\boldsymbol{\omega}}$, the angular acceleration vector, as a scalar position-invariant quantity that is approximately invariant to position shifts of the sensor unit on a given rigid body part but typically changes with time during motion

$$\mathrm{p} \triangleq \mathbf{a} \cdot \frac{\dot{\boldsymbol{\omega}}}{|\dot{\boldsymbol{\omega}}|}. \tag{1}$$

In our notation, a dot between two vectors denotes the inner product of the two vectors and a dot above a vector indicates the first derivative of that vector.

Another feature not influenced by the positioning of the sensor unit on the body is sensor-unit orientation with reference to the Earth frame E. Throughout this article, we estimate the 3-D orientation of the sensor unit at a given time sample using the algorithm we newly developed and presented in [42] that

processes the instantaneous sensory data. The algorithm relies on the efficient representation of 3-D orientation using a 4×1 quaternion $\mathbf{q}$ and provides higher activity recognition accuracy compared to state-of-the-art methods, with much shorter computation time than most of them.

The unit quaternion $\mathbf{q} = [q_1, q_2, q_3, q_4]^{\mathrm{T}}$ with four elements provides a convenient mathematical notation for representing orientations and rotations of objects in 3-D. Since a unit quaternion has an Euclidean norm of unity ($|\mathbf{q}| = 1$), only three of its elements are independent. Compared to rotation matrices (with nine elements), quaternions are more compact, numerically more stable, and faster to compute. Unlike Euler angles, they are simpler to compose and not susceptible to the "gimbal lock" problem caused by singularities [43].

Thus, in our recent work, which focused on achieving position invariance on individual rigid body parts for activity recognition, we investigated various arrangements of the position-invariant time sequences $\boldsymbol{\omega}$, $\mathbf{m}$, $|\boldsymbol{\omega}|$, $|\mathbf{m}|$, p, and $\mathbf{q}$ [20]. We assessed their performances and identified those with superior performance when a sensor unit was randomly offset from its ideal position on a certain rigid body part. The combinations $\boldsymbol{\omega}\mathbf{m}$p and $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$ provided the most favourable results in terms of classification accuracy and computational cost [20]. These sequence combinations are simply the concatenation of the vector or scalar quantities that appear in the expressions. In this article, we first consider position invariance among *different* rigid body parts through the random interchangeability of sensor units. This is followed by the usage of only a single unit on any one of the body parts to classify the activities. Afterwards, we combine the two ways of achieving position invariance on different body parts that we develop here with our earlier methodology to attain position invariance on the *same* rigid body part using the sequence combinations $\boldsymbol{\omega}\mathbf{m}$p and $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$. We finally incorporate our previously developed method on orientation invariance into this combined position-invariant scheme.

III. ACHIEVING POSITION INVARIANCE THROUGH INTERCHANGEABLE SENSOR UNITS

A. SVD-Based Method for Interchangeable Sensor Units

Interchanging the sensor units causes a permutation of the axes corresponding to the time-domain data derived from the different sensor units. If this interchange in sensor units is not tracked and countered, activity recognition accuracy suffers greatly. In this section, we propose a method that allows the user the freedom to randomly interchange the sensor units with minimal effect on activity classification performance.

The proposed method uses the SVD to generate an orthogonal transform for matrix factorization of the measurement data, derived from the sensor units. The data have underlying features that are typically linearly correlated. The SVD converts this correlated data into a set of uncorrelated principal components (axes) which accurately represents the time evolution of the data. Moreover, projections of the data onto the principal axes are invariant to row permutations in data matrix, caused by, possibly random, interchanging of the sensor units. This property importantly manifests as robustness to
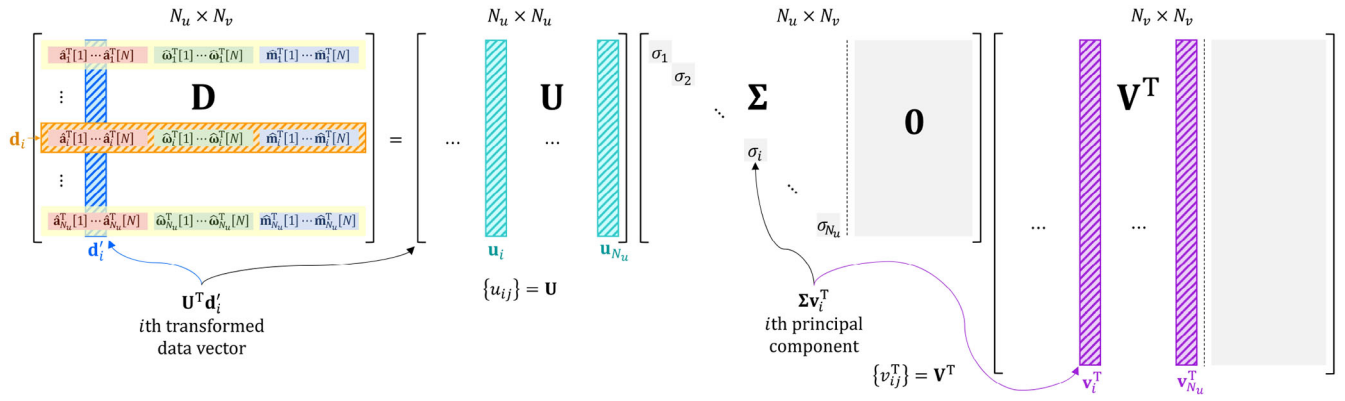
Fig. 1. Graphical illustration of the compact SVD operating on one segment of the acquired data to allow random interchangeability of the sensor units.

sensor-unit interchanging. Our approach for achieving position invariance can be viewed as analogous to that adopted in [21] for attaining sensor-unit orientation invariance.

Consider that the 3×1 column vectors $\mathbf{a}_i[n]$, $\boldsymbol{\omega}_i[n]$, and $\mathbf{m}_i[n]$ contain a snap-shot (sample), at time n, $n = 1, \ldots, N$, of the x, y, z coordinate measurements from the ith sensor unit, $i = 1, \ldots, N_u$, comprising the accelerometer, gyroscope, and magnetometer, respectively. Furthermore, consider that these vectors are normalized to have unit variance. The normalization is indicated by a hat ($\hat{\ }$) symbol as in $\hat{\mathbf{a}}_i[n] = [\hat{a}_{x_i}[n], \hat{a}_{y_i}[n], \hat{a}_{z_i}[n]]^{\mathrm{T}}$, where $(\cdot)^{\mathrm{T}}$ denotes the matrix transpose. Then, for every N-sample segment of the acquired, normalized, sensor-unit data, our approach consists of the following steps.

1) *Data Arrangement:* Now, suppose that N samples of the vectors $\hat{\mathbf{a}}_i[n]$, $\hat{\boldsymbol{\omega}}_i[n]$, and $\hat{\mathbf{m}}_i[n]$, for the ith sensor unit are concatenated into the row vector

$$\mathbf{d}_i = \Big[\hat{\mathbf{a}}_i^{\mathrm{T}}[1] \ldots \hat{\mathbf{a}}_i^{\mathrm{T}}[N], \hat{\boldsymbol{\omega}}_i^{\mathrm{T}}[1] \ldots \hat{\boldsymbol{\omega}}_i^{\mathrm{T}}[N],$$
$$\hat{\mathbf{m}}_i^{\mathrm{T}}[1] \ldots \hat{\mathbf{m}}_i^{\mathrm{T}}[N] \Big] \quad (2)$$

where $\mathbf{d}_i \in \mathbb{R}^{1 \times N_v}$ and $N_v = 3 \times 3 \times N = 9N$. Considering that there are N_u sensor units, with $N_u < N_v$, then a data matrix $\mathbf{D} \in \mathbb{R}^{N_u \times N_v}$ containing the segment of measurement data from all the sensor units can be constructed by stacking the row vectors $\mathbf{d}_i$, thus

$$\mathbf{D} = \big[\mathbf{d}_1^{\mathrm{T}} \, \mathbf{d}_2^{\mathrm{T}} \, \ldots \, \mathbf{d}_{N_u}^{\mathrm{T}} \big]^{\mathrm{T}}. \quad (3)$$

Note that when the sensor units are randomly interchanged, the rows of the data matrix $\mathbf{D}$ are permuted accordingly.

2) *Matrix Decomposition:* The data matrix $\mathbf{D}$ in (3) can be factored into a product of orthogonal matrices, namely, $\mathbf{U} \in \mathbb{R}^{N_u \times N_u}$ and $\mathbf{V} \in \mathbb{R}^{N_v \times N_v}$, and a diagonal matrix, $\boldsymbol{\Sigma} \in \mathbb{R}^{N_u \times N_v}$, by way of the SVD; that is

$$\mathbf{D} = \mathbf{U} \, \boldsymbol{\Sigma} \, \mathbf{V}^{\mathrm{T}} \quad (4)$$

where $\mathbf{U}^{-1} = \mathbf{U}^{\mathrm{T}}$ and $\mathbf{V}^{-1} = \mathbf{V}^{\mathrm{T}}$. Since $\mathbf{D}$ has only $N_u < N_v$ rows, its rank is at most N_u. Hence, assuming then a rank of N_u, the factorization in (4) can be

expressed as the compact (or economical) SVD [44]

$$\mathbf{D} = \mathbf{U} \big(\boldsymbol{\Sigma}_{N_u} \quad \mathbf{0} \big) \begin{pmatrix} \mathbf{V}_{N_u}^{\mathrm{T}} \\ \mathbf{V}_0^{\mathrm{T}} \end{pmatrix} = \mathbf{U} \, \boldsymbol{\Sigma}_{N_u} \, \mathbf{V}_{N_u}^{\mathrm{T}} \quad (5)$$

where $\mathbf{0} \in \mathbb{R}^{N_u \times (N_v - N_u)}$ is the zero matrix and $\boldsymbol{\Sigma}_{N_u} \in \mathbb{R}^{N_u \times N_u}$ is a diagonal matrix containing the nonzero singular values only, which are ordered as

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{N_u} > 0. \quad (6)$$

The columns of $\mathbf{U}$ and $\mathbf{V}_{N_u} \in \mathbb{R}^{N_v \times N_u}$ are the respective left and right singular vectors of $\mathbf{D}$. The columns of $\boldsymbol{\Sigma}_{N_u}$ and $\mathbf{V}_{N_u}$ constitute an exact representation of $\mathbf{D}$; and can provide the best rank N_u approximation in terms of the Frobenius norm [44].

3) *Data Transformation:* The principal components corresponding to the data in $\mathbf{D}$ are given by

$$\mathbf{W} = \mathbf{U}^{\mathrm{T}} \mathbf{D} = \boldsymbol{\Sigma}_{N_u} \, \mathbf{V}_{N_u}^{\mathrm{T}} \quad (7)$$

where $\mathbf{W}$ has the same dimensions as $\mathbf{D}$. In other words, the transformed signals are a weighted linear combination of the sensor-signal snapshots, namely, $\mathbf{d}_i'$, which are the columns of $\mathbf{D}$. Intuitively, the transformed signals may also be viewed as projections of $\mathbf{d}_i'$ onto the N_u-dimensional principal axes.

4) *Transformed-Data Reorganization:* The transformed data in $\mathbf{W}$ are partitioned according to the three sets of measurement data in (2). In effect, we reverse the concatenation operations involved in step 2 so that the transformed data have the same relevance as the raw measurements, and so can be fed to the standard activity recognition system without the need for further processing.

The four steps of the algorithm outlined above are executed on a block-by-block basis. Specifically, the sensor data are first split into short segments; the SVD and thus the principal components are computed for each segment, independent from all other segments. This process is illustrated, for one segment of the acquired data, in Fig. 1. In this way, our method is able to accommodate for a random rearrangement of the sensor units each time the system considers a new segment of motion data, both during training and testing. Note that the principal components are calculated for the training, as well as the test data,

for activity classification. This enables a matching between the two types of data, and hence, improves activity recognition accuracy.

B. Interchangeability Coupled With Position Invariance on a Given Rigid-Body Part

In a previous investigation of the invariance to position changes on the same rigid-body part [20], [24], we have employed two different sequence combinations, $\boldsymbol{\omega}\mathbf{m}$p and $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$, involving $\boldsymbol{\omega}$, $\mathbf{m}$, p, and possibly $\mathbf{q}$, with the respective dimensions $3 \times 1, 3 \times 1, 1 \times 1$, and 4×1.

To allow random interchangeability of the sensor units and simultaneously attain invariance to position shifts on a given rigid body part, we apply SVD on one of the two sequence combinations. Note that the sequence combination transformed by SVD is no longer $\boldsymbol{\omega}\mathbf{m}\mathbf{a}$, directly acquired from the three different sensor types in each unit. Instead, we replace the sequence $\mathbf{a}$ with either p or p$\mathbf{q}$ derived from the recorded sensory data to obtain either $\boldsymbol{\omega}\mathbf{m}$p or $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$. We handle the modified sequence combinations in the same way as we treat the raw sequence combination $\boldsymbol{\omega}\mathbf{m}\mathbf{a}$ when applying the SVD as described in Section III-A (except for the difference in dimensionality).

C. Interchangeability Coupled With Placement Invariance on a Given Rigid-Body Part

Using the SVD allows random interchangeability of the sensor units affixed to different rigid body parts (see Section III-A). We have also combined it with our former approach to additionally achieve position invariance as the sensor units are shifted while still on the same rigid body part in Section III-B. In this section, we extend the flexibility yet further by permitting arbitrary placement of the sensor units at any orientation as well. In doing so, we can achieve invariance to both position and orientation simultaneously while maintaining a good level of accuracy. To this end, the same SVD used to enable interchangeable units is applied to the modified sequence combination $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$, proposed earlier in [20] and [24] to achieve position as well as orientation invariance. Note that the first two members of this sequence combination are expressed in the Earth frame E, the third member is a scalar quantity, and the last member $\delta\mathbf{q}$ is the differential quaternion. To obtain this modified sequence combination, we first need to estimate the 3-D orientation of the sensor unit with reference to frame E based on the sensor readings [42]. Employing the method in [42], the first two sequences of the sequence combination $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$, originally acquired in the sensor unit frame, can be transformed to frame E. Furthermore, we use the *differential orientation quaternion* $\delta\mathbf{q}$ that represents changes in orientation in place of the orientation quaternion $\mathbf{q}$. In calculating $\delta\mathbf{q}$, we first compute a measure of the sensor-unit-frame rotation, namely, the differential rotation matrix $\mathbf{R}_n$, relative to frame E between successive sample times, n and $n + 1$ [22]. Using a 3×3 matrix (with nine elements) to represent a 3-D rotational transformation lacks efficiency since any 3-D rotation can be expressed in terms of three Euler angles. However, such three-angle representation has

a singularity problem; therefore, we represent the rotational transformation described by the differential rotation matrix $\mathbf{R}_n$ compactly by a 4×1 differential quaternion $\delta\mathbf{q}_n$ with reference to frame E [45]. Thus, given the elements of $\mathbf{R}_n$, $r_{ij}\ i,j = \{1, 2, 3\}$,

$$\delta\mathbf{q}_n = \begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \delta q_4 \end{bmatrix} = \begin{bmatrix} \dfrac{\sqrt{1 + r_{11} + r_{22} + r_{33}}}{2} \\ \dfrac{r_{32} - r_{23}}{4\sqrt{1 + r_{11} + r_{22} + r_{33}}} \\ \dfrac{r_{13} - r_{31}}{4\sqrt{1 + r_{11} + r_{22} + r_{33}}} \\ \dfrac{r_{21} - r_{12}}{4\sqrt{1 + r_{11} + r_{22} + r_{33}}} \end{bmatrix}. \quad (8)$$

For brevity's sake, we have not shown the dependence of the elements of $\mathbf{R}_n$ and $\delta\mathbf{q}_n$ on n here. Finally, we remove the subscript n in $\delta\mathbf{q}_n$ as well, and simply use $\delta\mathbf{q}$ in what follows.

Once the sequence combination $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$ is obtained from the recorded data, we apply the SVD to this sequence combination with the respective dimensions 3×1, 3×1, 1×1, and 4×1 in the first step of the approach outlined in Section III-A.

IV. ACTIVITY CLASSIFICATION WITH A SINGLE UNIT

A. Three Variants of Single Unit-Based Classification

Another flexibility and simplification that the user of the wearable system can benefit from is to recognize activities processing the data recorded by only a single sensor unit. In this scenario, the activity recognition system is trained with sensory data collected from multiple sensor units worn on different rigid body parts (head, torso, and lower/upper limbs). Data acquired from all of the available positions are used for training. However, activities are recognized based on the test data acquired from only a *single* sensor unit that is affixed to any one of these body parts. Once the training of the system is completed, this approach provides the user the pliability and freedom to affix the unit to any preferable or convenient body part. In the test phase, the user does not need to provide any information to the activity recognition system about to which of these multiple positions the unit is affixed. In the following, we will consider three variants of a methodology that we will refer to as single unit-based classification (SUBC).

SUBC₁: Here, we adopt a generalized classifier for training the first variant. Consider the column vector $\mathbf{f}_i^{(j)}$ that contains the features extracted from the jth data segment derived from the ith sensor unit, for $i \in \{1, \ldots, N_u\}$ and $j \in \{1, \ldots, N_s\}$, where N_s denotes the total number of data segments used for training. We consider the collection of features extracted from each unit as a distinct training instance and construct the training set as

$$\mathscr{T}_{\mathrm{SUBC}_1} = \Big[\mathbf{f}_1^{(1)}, \mathbf{f}_2^{(1)}, \ldots, \mathbf{f}_{N_u}^{(1)}, \quad \mathbf{f}_1^{(2)}, \mathbf{f}_2^{(2)}, \ldots, \mathbf{f}_{N_u}^{(2)},$$
$$\ldots, \quad \mathbf{f}_1^{(N_s)}, \mathbf{f}_2^{(N_s)}, \ldots, \mathbf{f}_{N_u}^{(N_s)}\Big]. \quad (9)$$

On the other hand, consider stacking the features related to all sensor units in a vector, thus

$$\mathbf{f}^{(j)} = \left[\left(\mathbf{f}_1^{(j)}\right)^{\mathrm{T}}, \ \left(\mathbf{f}_2^{(j)}\right)^{\mathrm{T}}, \ldots, \left(\mathbf{f}_{N_u}^{(j)}\right)^{\mathrm{T}}\right]^{\mathrm{T}}. \quad (10)$$

The training set $\mathscr{T}_{\mathrm{ref}} = [\mathbf{f}^{(1)}, \mathbf{f}^{(2)}, \ldots, \mathbf{f}^{(N_s)}]$, comprising N_s training vectors, serves as our reference case.

In both cases, the training feature vectors have true labels associated with the activities to which they belong, so that there are as many classes as the number of activities, N_a. Thus, in the generalized classifier, we have $N_u \times N_s$ feature vectors for training, which are N_u times more in number and N_u times smaller in size than in the reference case.

Through the use of the generalized classifier, we categorize the test vectors and assign to appropriate activity classes in the test phase based on only one sensor unit; hence, we have distinct test feature vectors associated with each unit as in the training set. This is done without needing to know to which body part the unit was affixed. Because the training set comprises feature vectors associated with all the sensor unit positions from which data were acquired, we expect the classifier to match the given test feature vector extracted from a single sensor unit position with these training feature vectors. This is not a trivial task due to the variability of the data within the activity classes since different participants tend to perform the activities in their own way [46].

SUBC$_2$: The second variant involves a two-stage activity recognition procedure [35].

In the first stage, we classify the position of the sensor unit into one of the positions used for acquiring the data set. We adhere to the same classification scheme as in activity recognition, with the number of classes being equal to N_u instead of N_a.

In the second stage, we train as many activity classifiers as the number of sensor unit positions N_u, where each activity classifier is specialized on a particular unit position. The set of training vectors for the *i*th unit position is constructed as

$$\mathscr{T}_{\mathrm{SUBC}_2, i} = \left[\mathbf{f}_i^{(1)}, \mathbf{f}_i^{(2)}, \ldots, \mathbf{f}_i^{(N_s)} \right], \quad i \in \{1, \ldots, N_u\}. \quad (11)$$

Making use of the unit position identified in the first stage, we deploy the activity classifier trained for that particular position to recognize the activity.

SUBC$_3$: In the third variant, we attempt to classify the sensor unit position and the activity type at the same time [39] by considering each unit position and activity combination as a distinct class. The number of classes becomes $N_u \times N_a$. We form the training set in the same way as in SUBC$_1$, see (9), except that in this case, the training labels contain both the sensor unit position and the activity type [see the class labels in the legend of Fig. 5(c)].

B. Single Unit Usage Coupled With Position Invariance on a Given Rigid-Body Part

We apply each of the three variants of SUBC by extracting features from either $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$ or $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$ to gain invariance to position shifts as the sensor unit is moved on the surface of the same rigid body part. This scheme requires to place only a single sensor unit anywhere on any one of the rigid body parts (at proper orientation) on which a sensor unit was placed and data were acquired during the training phase.

TABLE I
INVARIANCE AND INTERCHANGEABILITY PROPERTIES OF THE PROPOSED DATA TRANSFORMATION TECHNIQUES. PI: POSITION INVARIANCE AND OI: ORIENTATION INVARIANCE

data transformation technique	PI on the same rigid body part	PI among different body parts	OI	interchange-ability of sensor units
$\boldsymbol{\omega}\mathbf{m}\mathbf{a}$ (raw sequences) no transformation	✗	✗	✗	✗
$\boldsymbol{\omega}\mathbf{m}\mathrm{p}$	✔	✗	✗	✗
$\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$	✔	✗	✗	✗
$\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}$	✔	✗	✔	✗
$\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$	✔	✗	✔	✗
SVD on $\boldsymbol{\omega}\mathbf{m}\mathbf{a}$	✗	✗	✗	✔
SVD on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$	✔	✗	✗	✔
SVD on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$	✔	✗	✗	✔
SVD on $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$	✔	✗	✔	✔
SUBC on $\boldsymbol{\omega}\mathbf{m}\mathbf{a}$	✗	✔	✗	*
SUBC on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$	✔	✔	✗	*
SUBC on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$	✔	✔	✗	*
SUBC on $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$	✔	✔	✔	*

* Not applicable because only a single sensor unit is used.

C. Single Unit Usage Coupled With Placement Invariance on a Given Rigid-Body Part

We apply the three variants of SUBC that are detailed in Section IV-A on the sequence combination $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$, which can achieve position as well as orientation invariance on a given rigid body part. This allows classifying the activities by processing the data acquired from only a single sensor unit that is affixed to any position and orientation on any one of the rigid body parts from which data were acquired during the training phase.

Table I summarizes the invariance and unit-interchangeability properties of the proposed data transformation techniques described in Sections III and IV.

V. DATA SET AND THE ACTIVITY RECOGNITION METHODOLOGY

A. Data set

To assess the performance of our activity recognition system with invariance property to the positioning of the units, we processed the activity data set that was recorded by our research team and that is publicly available [23], [47]. In the experiments, eight voluntary participants wore five Xsens MTx motion-sensor units [48] at the center of their chest, on the right and left wrist, and on the outer sides of right and left knee, as depicted in Fig. 2(a). The orientations of the axes of each sensor unit are illustrated in Fig. 2(b). We received the approval of the Local Ethics Committee of Bilkent University to conduct experiments with human participants. The participants were healthy subjects, free from any movement disorders, and submitted their written informed consent prior to the experiments. The physical attributes (profiles) of the participants are provided in Table II.

TABLE II
PROFILES OF PARTICIPANTS IN THE EXPERIMENTS

participant index	gender	age (yrs)	height (m)	weight (kg)
1	F	25	1.70	63
2	F	20	1.62	54
3	M	30	1.85	78
4	M	25	1.82	78
5	M	26	1.83	77
6	F	23	1.65	50
7	F	21	1.67	57
8	M	24	1.75	75
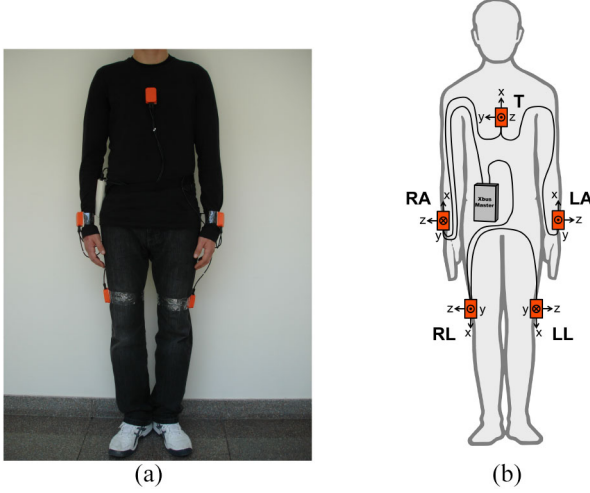
Fig. 2. (a) Placement and (b) wiring of the body-worn motion-sensor units. (The body sketch in (b) is from http://www.clker.com/clipart-male-figure-outline.html; the motion-sensor units, their wiring, and the Xbus Master were added by the authors.)

The experiments were conducted in an outdoor parking lot, possessing a flat, level surface, in the Electrical and Electronics Engineering Building, and at the Physical Education and Sports Center, all within the confines of the Bilkent University campus. Each participant performed 19 types of daily and sports activities, each for 5 min. They were not given any specific instructions, and were free to perform these activities in their own way, as naturally as possible.

> Sitting (A_1); standing (A_2); supine (A_3); reclined-right side (A_4); climbing (A_5) and descending stairs (A_6); in an elevator, upright & still (A_7) and walk-ing (A_8); pacing in the car park (A_9); 4 km/h walking pace on a flat treadmill (A_{10}); 4 km/h walking pace on a 15°-inclined treadmill (A_{11}); 8 km/h running pace on a level treadmill (A_{12}); exercising on a: stepper (A_{13}), cross trainer (A_{14}), bicycle in horizon-tal (A_{15}) and vertical (A_{16}) positions; rowing (A_{17}); jumping (A_{18}); playing basketball (A_{19}).

The activities listed above can be divided into two broad categories: 1) static and 2) dynamic. Postures, such as sitting, standing, or lying down, belong to the first category, in which the body limbs remain static. Dynamic activities can be further divided into two categories, namely, quasiperiodic and erratic, where some body limbs move with almost periodic and non-periodic patterns, respectively. Thus, depending on the activity type and the subject's personal style, the recorded motion pat-tern at a given limb can be static, quasiperiodic, or erratic

dynamic. Among the listed activities, we consider A_1–A_4 as static activities, A_7, A_8, and A_{19} to be erratic dynamic, and all the remaining ones as quasiperiodic activities. Depending on the way the jumping activity is performed, A_{18} can be quasiperiodic or erratic dynamic. In our data set, it is quasi periodic.

Given that three triaxial sensors (an accelerometer, a gyro-scope, and a magnetometer) are enclosed in each body-worn sensor unit, the data recorded consisted of 45 ($=5$ sensor units $\times$ 3 sensor types $\times$ 3 dimensions) time-domain sequences for each activity performed by each subject using a uniform sampling frequency of 25 Hz. Since each subject performed each activity for 5 min, initially, 5-min recordings were made, each containing 7500 ($=5$ min $\times$ 60 sec/min $\times$ 25 samples/sec) time samples.

B. Methodology for Activity Classification

The standard activity classification scheme, which we have followed here, involves the basic stages of segmentation of the data, extraction, normalization, and reduction of the features, and the classification based on (possibly transformed) data [9], [11], [22]. This scheme has been employed in various studies. For example, in [8], a similar procedure was used to the one employed here, the differences being: 1) a 50% overlap in the segments of 2 s is considered; 2) a smaller set of features is extracted from the time-domain signals; and 3) five types of rehabilitation exercises are recognized by a single random forest (RF) classifier, trained with 66% percentage split. In this study, we partition the 5-min recordings into 5-s long segments that do not overlap. Using the segmented data in its original, unprocessed form corresponds to the reference case. As described in Sections III-A and IV-A, respectively, the segmented data can also be processed using the SVD or SUBC to gain an effective level of invariance to sensor-unit positioning on body parts.

The following statistical quantities were extracted from the time series corresponding to each segment of data: the two extrema; mean and variance; autocorrelation-function terms corresponding to the ten lags $5, 10, 15, \ldots, 50$; skewness and kurtosis; and the five largest values and corresponding frequencies of the discrete Fourier transform, with frequency separation of at least 11 samples. We extracted 26 features from each data segment. For each segment, the feature vec-tors comprised the concatenation of the 26 features from five sensor units having nine axes, resulting in 1170-element fea-ture vectors. These vectors were used with the reference case, which processes the raw sequence combination **ωma** without aiming to enforce position invariance.

Note that the number of (feature) elements of a feature vector is determined by the number of scalar values and vec-tor entries corresponding to a sequence combination. For the sequence combination **ωm**p, the number of features in each feature vector was 910 ($=5$ sensor units $\times$ 7 dimensions $\times$ 26 features), and for **ωm**pq, it was 1430 ($=5$ sensor units $\times$ 11 dimensions $\times$ 26 features). Transforming the sequences to frame E does not change the size of the feature vector. Principal component analysis (PCA) [49] was employed to reduce the feature space to that corresponding to the 30 most

dominant features, where data from each subject were normalized such that values fell between 0 and 1, inclusive. This is an orthogonal transformation, which aims to identify the most discriminative features and sorts the variances of the transformed features in descending order.

We use the following machine learning algorithms to classify activities.

Support Vector Machines (SVMs): With use of the well-known Gaussian radial basis function (RBF) kernel, $\mathcal{G}_{\text{RBF}}(\mathbf{f}_1, \mathbf{f}_2) = e^{-\gamma \|\mathbf{f}_1 - \mathbf{f}_2\|^2}$ for any two reduced feature vectors $\mathbf{f}_1$ and $\mathbf{f}_2$, the feature space is mapped to a higher-dimensional space to be partitioned into hyperplanes. Two parameters are involved. Jointly optimized via a two-level grid search, the optimal values of the kernel γ and penalty C parameters [see (1) in [50]], $C = 5$ and $\gamma = 0.1$, are employed for this classifier in the remainder of this article. After training a binary SVM for each class pair, the highest confidence level is taken for the decision [51]. We used the MATLAB toolbox LibSVM [52] to implement the SVM classifier.

Artificial Neural Networks (ANNs): Our multilayered ANN consists of three layers of neurons where each neuron is activated by a sigmoid-type activation function. The number of neurons in the: 1) input (first) layer is 30, i.e., the reduced number of features; 2) output (third) layer is $K = 19$, the number of classes; and 3) hidden (second) layer is given by $\lfloor < \log(2K)/\log 2, 2K-1 > \rceil$ where $< ., . >$ denotes the sample average of its two arguments and floor/ceiling operation on the average fetches the nearest integer. The first argument of the averaging operator corresponds to hyperplanes intersecting at different positions (optimistic case) and the second to hyperplanes in parallel (pessimistic case). Starting from a random initialization in the $[0, 0.2]$ interval, while training, the weights and bias defining interconnecting neurons are tuned via backpropagation [53] with a learning rate of 0.3. Once the error falls by 0.01 below the mean error, taken over the last ten epochs, the algorithm terminates. The class decision is given by the dominant neuron—indicated by a scalar quantity—when fed with a test feature vector during the test phase.

Bayesian Decision Making (BDM): A BDM was setup with feature vectors drawn from a multivariate Gaussian distribution. The training feature vectors, for each class, were approximated by arbitrary-covariance, multivariate Gaussian distributions when in the training phase. Maximum likelihood estimation (MLE) was employed to estimate the covariance matrix and mean vector from the training vectors; and classification was performed based on the maximum *a posteriori* decision rule. The mean vector and covariances were computed via the MLE as, respectively, the arithmetic mean of the feature vectors and sample covariance matrix. In the testing phase, a conditional probability was computed from each class's test vector. Utilizing the maximum *a posteriori* decision rule, the activity class associated with the maximum conditional probability was chosen [49], [54].

Linear Discriminant Classifier (LDC): The main difference of LDC from BDM is that in LDC, the average of the covariance matrices, individually calculated for each class, is used overall. Hence, the normal distributions modeling the distribution of the feature vectors in each class have identical covariance matrices but different mean vectors so that they are centered at different points in the feature space. Thus, the decision boundaries in the feature space correspond to hyperplanes, permitting linearly separable activity classes [54].

k-Nearest Neighbor (k-NN): The sole requirement for the training of the k-NN classifier is to store the training instances (feature vectors) together with their class labels. The selection of the parameter k is important since it corresponds to the number of nearest neighbors of a given test vector to use. We have considered the range $1 \leq k \leq 30$ and selected $k = 7$. During the test phase, we identified those k training vectors with the smallest Euclidean distance to the test vector [54]. The class decision is reached by identifying the class that the majority of these k vectors belong to [54].

Random Forest (RF): Using multiple decision trees, an RF classifier can be obtained [55] where training of each tree is done by random and independent sampling of the training data. *Normalized information gain* is used as the splitting criterion at each node. Class decision is based on majority voting over the tree decisions. We have employed 100 decision trees and noted that increasing the number does not boost the classification rate significantly, while increasing the computational load considerably.

Orthogonal Matching Pursuit (OMP): The training phase is very simple and exactly the same as in k-NN. During testing, however, the OMP algorithm [56] is used in an iterative manner to create a *sparse representation* of every test signal, each of which is a composite of vectors from an extremely small subset of the training set that adhere to a desired error bound. At each iteration, an additional training signal is chosen until the targeted error level for the representation (10^{-3}) is attained at termination. The class associated with the smallest residual error is selected. This minimized residual error is obtained by selecting the composite test signal, generated as a linear combination of the training signals, which gives rise to smallest reconstruction error, for each class.

Cross-Validation Techniques: We have employed two cross-validation techniques, P-fold and leave-one-subject-out (L1O), to evaluate the outputs of the classifiers. In P-fold, we randomly segment the dataset into $P = 10$ equal partitions. We classified the feature vectors in each partition, deploying one of the seven machine learning classifiers trained by the feature vectors in the rest of the partitions, and calculated the average of the accuracies for each partition to get the overall accuracy. In L1O, we performed the partitioning in a subjectwise manner (instead of randomly) so that the data in each partition belong to only one of the eight subjects [54]. This means that there were eight partitions in L1O at each round, the feature vectors of a given subject were omitted while the classifier was trained with the remaining subjects' feature vectors. We then employed the omitted subject's feature vectors during the classification (testing) stage. We repeated the same procedure for every subject and averaged out the results. L1O cross validation is more demanding and highly sensitive to the variability in the data between the subjects because the training and test
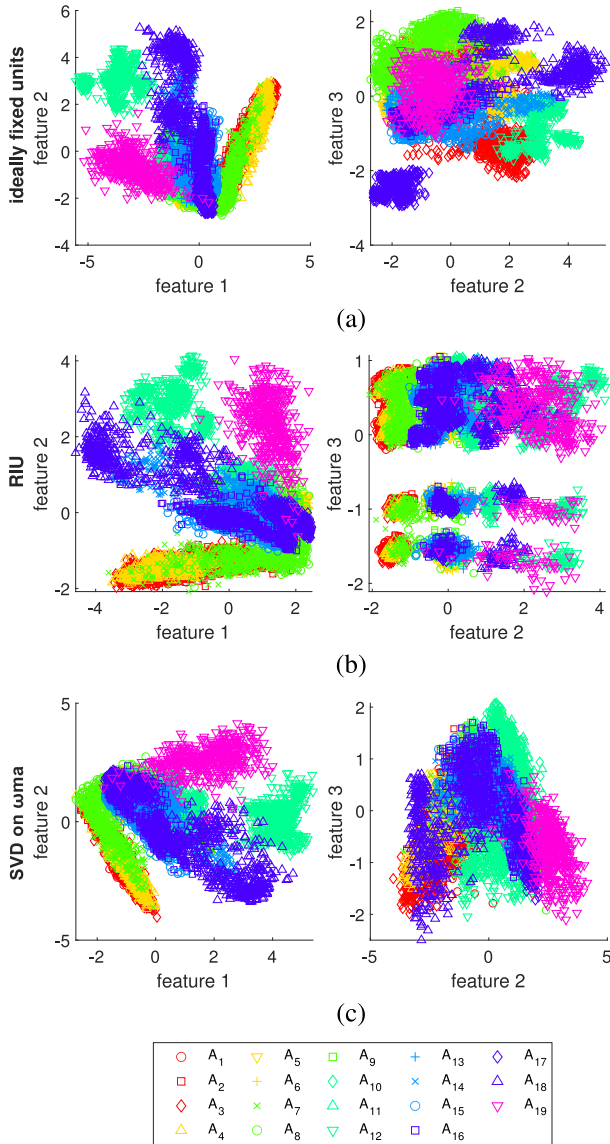
Fig. 3. Pairwise scatter plots of the first three features selected by PCA for (a) ideally affixed units, (b) RIU, and (c) SVD operating directly on the originally acquired sequences **ωma**.

sets are associated with different subjects, typically with more variability among them [57]. L1O is often used for assessing the generalization capability of the classifier to subjects that the classifier was not trained for and preferred to *P*-fold in scenarios where it is not possible to acquire training instances from eventual end users of the system.

VI. RESULTS

In this section, two reference cases are of interest to compare the accuracy results of our proposed approaches.

1) *Ideally Affixed Units:* Each unit properly placed on the associated body part at its ideal position and orientation.
2) *Randomly Interchanged Units (RIU):* The units randomly interchanged among themselves and placed on body parts at ideal position and orientation.

The first case is an idealization, which is difficult to realize in practice. On the other hand, the second case is considered to represent a worst case scenario where the user completely

mixes up their units; however, this convenient situation is not desirable since the mixage of the units is not countered in any way to avoid accuracy degradation. In both cases, no counter-measures are taken to attain any kind of invariance (position and/or orientation) to deviations in sensor-unit placement from the ideal (on the same body part). Thus, both are highly vulner-able in this respect, which is expected to cause severe accuracy degradation. We have considered both cases for benchmark-ing to evaluate the performance of our proposed position (and orientation) invariance techniques.

Our SVD-based method, for countering unit interchange-ability, is evaluated with both cases 1) and 2); whereas the performance of our SUBC-based technique, for single unit usage, is assessed with case 1) only, since 2) is not relevant for SUBC.

A. Results of Interchangeable Sensor Units

1) Results of Randomly Interchanged Units: We randomly and independently *interchanged* the data from the five sensor units at the start of each 5-s segment of test data. When a test feature vector is extracted from the recordings of sensor units that are arranged differently than in the training phase, this renders into shuffling of the feature elements in the feature vectors. The activity recognition rate is expected to degrade considerably unless this is tracked and countered.

The pairwise scatter plots of the first three features, after feature reduction via PCA, are provided in Fig. 3. The high correlation between features 1 and 2 is striking for certain types of activity, in addition to the great degrees of over-lap/separation for certain others. Specifically, in Fig. 3(a), notice the significant overlap coupled with the pronounced elongation for activities A_{13}–A_{18}, which indicates consider-able similarity between the activities and the two features; this group, however, is clearly separated from, e.g., A_{19}, whose dull rounded shape is indicative of a low degree of similarity since it corresponds to activity that is erratic dynamic in nature, as defined in Section V-A. More interestingly, there is a notice-able reduction in the overall correlations and overlaps between features 2 and 3 in Fig. 3(b); e.g., there are clear separations between activities A_{13}–A_{18}.

The results for the reference case with ideally affixed units are provided in the topmost bars of each group of bars in Fig. 4, with 98.2% and 87.2% classification accuracy for *P*-fold and L1O cross validation. We provide the impact of RIU on the activity classification performance in the same figure in the second bars from the top of each group of bars. Randomly interchanging the units degrades the accuracy abruptly by 57.4% for *P*-fold and 52.8% for L1O cross validation com-pared to the reference case with ideally affixed units. This is directly attributed to the shuffling in the data due to randomly interchanging units and no precautions being taken to counter this with a view to alleviating accuracy degradation.

2) Results of SVD: Pairwise scatter plots of the first three features after feature reduction through PCA are provided in Fig. 3(c) for SVD operating directly on the unprocessed sequences **ωma**. Expectedly, we see that similarities between the activities A_{13}–A_{18} are preserved with SVD processing.
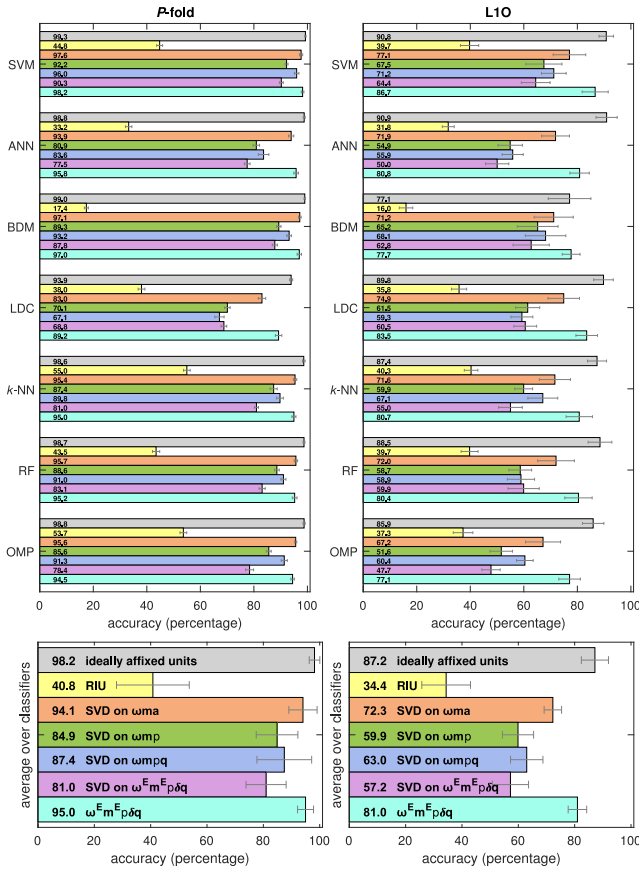
Fig. 4. Activity recognition accuracy for ideally affixed units, RIU, and the SVD operating on the sequence combinations $\boldsymbol{\omega}\mathbf{ma}$, $\boldsymbol{\omega}\mathbf{mp}$, $\boldsymbol{\omega}\mathbf{mpq}$, and $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$. The lengths of the bars indicate the accuracy values. The thin error bars represent ±1 standard deviation over the cross-validation iterations at the top and over the classifiers at the bottom part of the figure.

In Fig. 4, we provide the activity recognition accuracies comparatively for ideally affixed units (as described above), RIU, and SVD operating on various sequence combinations. The activity recognition rate for SVD operating on the directly acquired sequences $\boldsymbol{\omega}\mathbf{ma}$ corresponds to the third bar from the top for each group of bars in the figure. Unlike RIU, classification results were not significantly affected by the random interchanging of the sensor units when SVD was used to transform the sensory data. Hence, when the units are randomly interchanged, the accuracy is considerably improved with the use of SVD, compared to RIU, by 53.3% and 37.9% for *P*-fold and L1O cross validation, respectively. This improvement is due to the fact that the principal components from the SVD provide a basis for the data that is invariant to the interchanging/reordering of the units. It directly follows that working with the principal components, rather than the data itself, has a positive impact on activity recognition. Compared to the reference case with ideally affixed units, SVD processing on the originally acquired sequences $\boldsymbol{\omega}\mathbf{ma}$ results in a degradation of the accuracy to some extent: by 4.1% for *P*-fold and 14.9% for L1O cross validation. The reduction in accuracy is more than justified by the flexibility offered to the user (see Table I).

3) Results of SVD Coupled With Position Invariance on a Given Rigid-Body Part: SVD is applied to the sequences $\boldsymbol{\omega}\mathbf{mp}$ and $\boldsymbol{\omega}\mathbf{mpq}$ to attain position invariance as the position of an interchanged unit is shifted on a given rigid body part. From Fig. 4, it is clear that with both sequence combinations, we achieve accuracies close to that of using SVD on the originally recorded sequences $\boldsymbol{\omega}\mathbf{ma}$ with a small degradation in the accuracy for *P*-fold, but with a more noticeable drop for L1O cross validation. Specifically, compared to SVD operating on $\boldsymbol{\omega}\mathbf{ma}$, applying SVD on $\boldsymbol{\omega}\mathbf{mp}$ causes an accuracy reduction of 9.2% and 12.4% for *P*-fold and L1O, respectively. Employing SVD with $\boldsymbol{\omega}\mathbf{mpq}$ results in 6.7% and 9.3% respective degradation, for the same two cross-validation techniques. However, note that these accuracy levels are still above that of RIU by 44.1% and 25.5% for $\boldsymbol{\omega}\mathbf{mp}$ and by 46.6% and 28.6% for $\boldsymbol{\omega}\mathbf{mpq}$ for the two cross-validation techniques in respective order. Overall, the inclusion of the quaternion $\mathbf{q}$ in the sequence set brings an improvement to the results by enhancing the position-invariant trait of the sequence set. Therefore, it is possible to randomly interchange the sensor units and still achieve invariance to position shifts on a given rigid body part with a moderate degradation in the activity classification rate.

4) Results of SVD Coupled With Placement Invariance on a Given Rigid-Body Part: Fig. 4 also presents the results of stand-alone use of the sequence combination $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$ to attain position and orientation invariance on a given rigid body part, and SVD operating on $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$ for the same purpose when the units are randomly interchanged. The former, without random interchangeability of the units, yields 3.2% and 6.2% lower accuracy compared to the reference case with ideally affixed units. Allowing random unit interchangeability, for the case where SVD is applied to $\boldsymbol{\omega}\mathbf{ma}$ yields a degradation in accuracy as compared to the stand-alone use of $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$, even though the difference is not significant for *P*-fold (0.9%). Allowing both types of flexibility, i.e., random interchangeability of the units in combination with placement invariance on a certain rigid body part, through the use of SVD on $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$, further degrades the accuracy, as expected. In this case, the accuracy is 14.0% and 23.8% lower compared to the stand-alone use of $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$, 17.2% and 30.0% lower compared to ideally affixed units. Compared with RIU, where no countermeasures are taken to achieve any kind of invariance, these accuracy levels are 40.2% and 22.8% higher. In the last three comparisons, the two percentages reported in each case are for *P*-fold and L1O cross validation, respectively. We observe that the improvement in the accuracy with respect to RIU is less for L1O compared to *P*-fold. Note that when SVD operates on $\boldsymbol{\omega}^E \mathbf{m}^E \mathbf{p}\,\delta\mathbf{q}$, the sole requirement from the user is to affix only one sensor unit at any position and orientation on each one of the body parts on which sensor units were placed and training data were acquired during the training phase.

Note that in all three cases of using SVD above, i.e., in Section VI-A2 to Section VI-A4, the accuracy level of SVD stays considerably above that of RIU, the minimum difference being 40.2% and 22.8% for *P*-fold and L1O, respectively, in the last case. These accuracies can be considered favourable, given that, in the last case, position and orientation invariance was also achieved by the appropriate choice of a sequence set, which is unlike the case of RIU, where the units are ideally placed on the body parts without taking any precautions to achieve invariance.
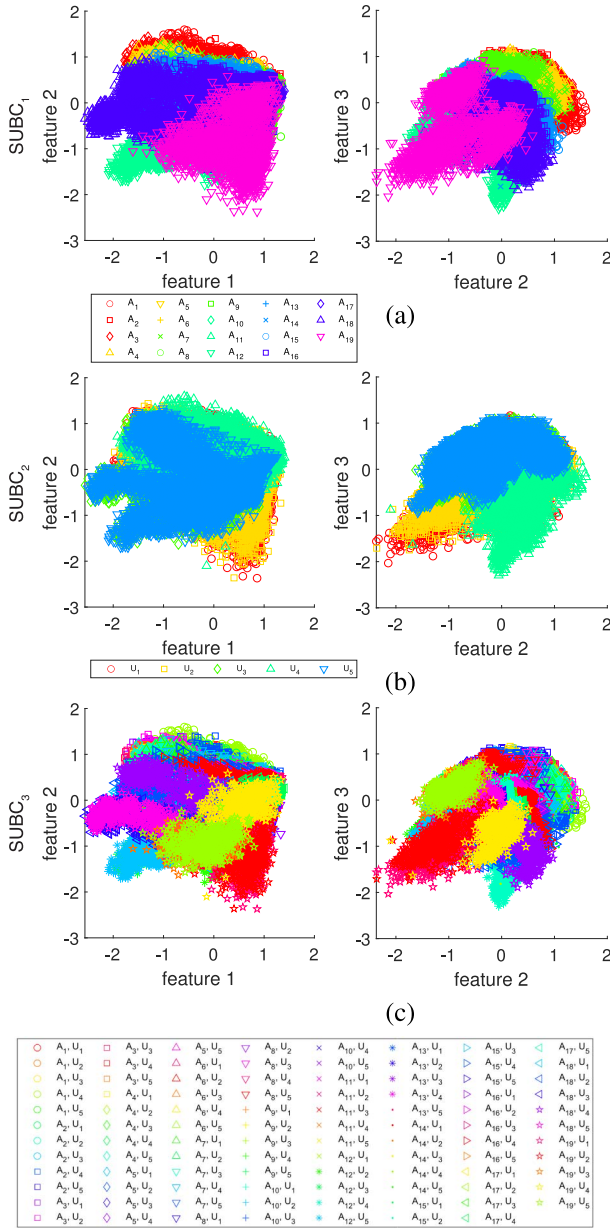
Fig. 5. Pairwise scatter plots of the first three features selected by PCA for (a) $SUBC_1$, (b) $SUBC_2$, and (c) $SUBC_3$. U_i: sensor unit i.

B. Results of Single Unit-Based Classification

1) Results of the Three Variants of SUBC: We provide the scatter plots of the first three features after feature reduction through PCA in Fig. 5 for the three variants of SUBC. The large overlap between the features suggests that feature correlations for the various activities are similar to those in Fig. 3. In the left-hand plots, there is mainly low to moderate correlation between the first two features, which is indicated by the circular, clumped nature of these scatter plots. It is clear that the features are segregated into more clusters with $SUBC_3$ than with both $SUBC_1$ and $SUBC_2$. This is mainly because there are five classes associated with each type of activity in $SUBC_3$.

In Fig. 6, the results for the reference case with ideally affixed units are provided again in the topmost bars of each
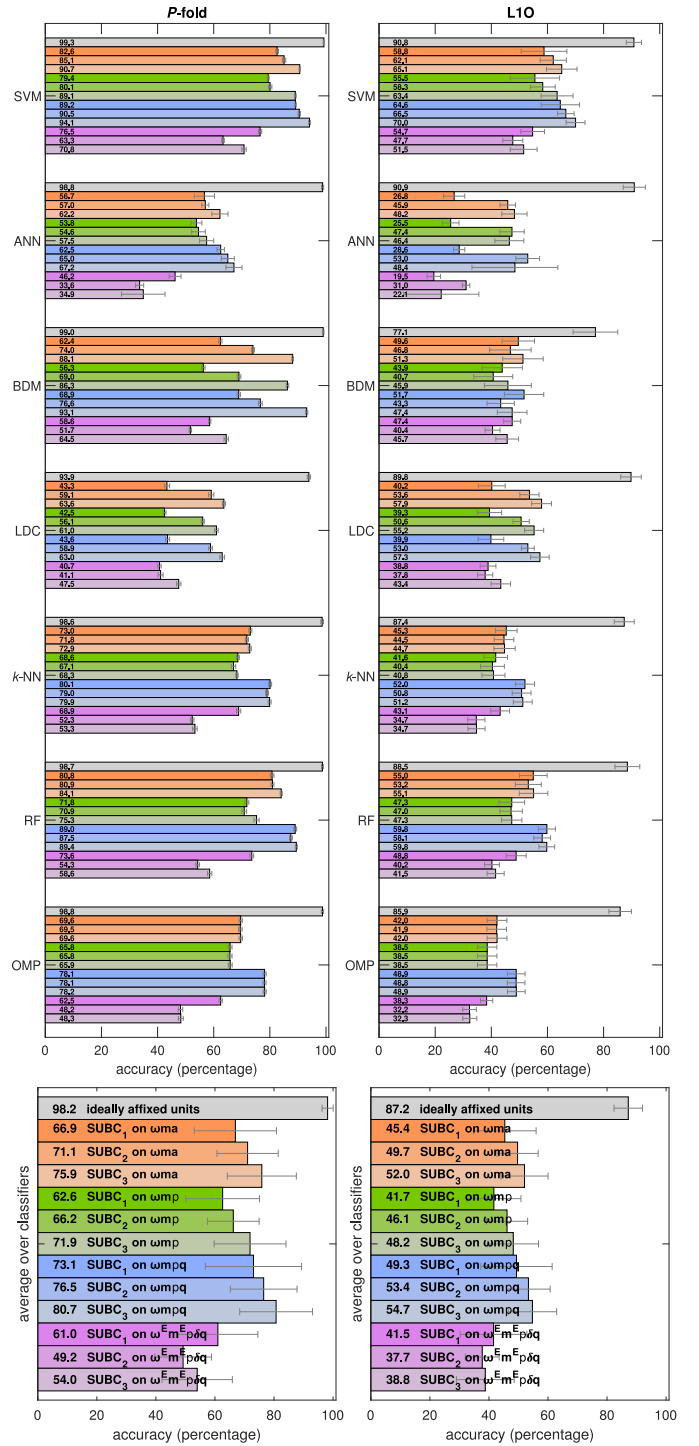

Fig. 6. Activity recognition accuracy for ideally affixed units and the three variants of SUBC employed on the sequence combinations $\boldsymbol{\omega}\mathbf{ma}$, $\boldsymbol{\omega}\mathbf{m}p$, $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$, and $\boldsymbol{\omega}^E\mathbf{m}^Ep\,\delta\mathbf{q}$.

group of bars. The bars underneath the topmost one are grouped into four sets of colors, each set comprising three shades of the color. In these bars, the activity recognition accuracies for the three variants of SUBC, namely, $SUBC_1$, $SUBC_2$, and $SUBC_3$, are compared. Each color group corresponds to the results of a different sequence combination where features are extracted from that particular sequence combination. The $SUBC_3$ is able to discern between activities very well despite it having to consider many more classes

TABLE III
Average Processing Time Needed to Transform the Raw Sequences $\boldsymbol{\omega}\mathbf{ma}$ to Obtain a New Representation per 5-s Time Segment

data transformation technique	run time per segment (ms)
$\boldsymbol{\omega}\mathbf{ma}$ (raw sequences) no transformation	—
$\boldsymbol{\omega}\mathbf{m}\mathrm{p}$	2.57
$\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$	11.19
SVD on $\boldsymbol{\omega}\mathbf{ma}$	18.00
SVD on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$	12.85
SVD on $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$	38.79
SVD on $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$	54.62
$\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}$	26.53
$\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}$	28.11
$\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$	32.86

($95 = 19$ activities $\times$ 5 units) compared to the other two variants. This is verified by the superior recognition performance of SUBC_3 in Fig. 6. Note that results pertaining to activity classification are presented here; the classification of sensor-unit positions is beyond the scope of this study. When any one of the three variants operate on the sequence combination $\boldsymbol{\omega}\mathbf{ma}$, we obtained activity recognition rates that were considerably lower than the reference case with ideally affixed units, as expected. We observe that among the three variants of SUBC, SUBC_3 obtains the highest accuracy, followed by SUBC_2, for both types of cross validation. Note that in this particular case, since the sequence combination $\boldsymbol{\omega}\mathbf{ma}$ containing the raw data is being processed, we do not aim to attain invariance to position and/or orientation regarding sensor-unit placement.

2) Results of SUBC Coupled With Position Invariance on a Given Rigid-Body Part: In Fig. 6, we also show the classification accuracies related to the three variants of SUBC in the case where a single unit is used together with the sequence combination $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$, for position invariance on a given rigid body part. In spite of the expected reduction in activity-recognition performance for this case, when compared to the reference case with ideally affixed sensor units, the use of $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$ permits flexibility in sensor unit positioning and tolerates deviations in position to an extent.

A striking result is the noticeable accuracy improvement gained with the three variants of the SUBC approach on the sequence combination $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$, even when compared with using each variant together with the sequences $\boldsymbol{\omega}\mathbf{ma}$. The former sequence combination removes the data components related to the position of the units on the rigid body parts to which they are affixed [20]. In this process, the information about the position of the units on the body as a whole may also be being partially removed, which may improve the accuracy when a single unit is used for classification. This effect may be more enhanced in the case of static activities where the body parts ideally do not move with respect to each other.

3) Results of SUBC Coupled With Placement Invariance on a Given Rigid-Body Part: In Fig. 6, we also show classification rates for the case where we employ the three SUBC variants together with placement (position as well as orientation) invariance on a given rigid body part. We observe that extracting features from the sequence combination $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$ significantly degrades the accuracy compared to the reference case with ideally affixed sensor units. The SUBC_1 variant performs the best when processing $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$, followed by SUBC_3. Despite being somewhat low, the achieved recognition rate is still considerably better (by nearly an order of magnitude) than randomized decision making, which has an average accuracy level of $1/19 = 5.26\%$ for 19 classes.

An overall observation related to the cross-validation process is that for all the different cases considered in this article, the classification accuracies obtained with L1O are lower than those obtained with *P*-fold. Furthermore, the reductions in accuracy are smaller in all cases of *P*-fold compared to L1O cross validation when compared to the reference case with ideally affixed units. (Note that both of these findings were also observed for the SVD approach in Section VI-A.) This difference between the two cross-validation techniques is mainly attributed to the subject-based partitioning of the data when using L1O. Since there is some variability in the way the different subjects perform the activities, the subjectwise partitions of L1O are less similar to each other compared to the randomized partitions of *P*-fold even for ideally affixed units. Combining data from different subjects in the same partition improves the classification accuracy by making the cross-validation process more robust to accuracy degradation. In contrast, including only the data acquired from a single subject in a given partition lowers the resulting accuracies.

VII. Runtime Analysis

Table III provides the average processing times per 5-s time segment to transform the originally recorded sequences $\boldsymbol{\omega}\mathbf{ma}$ into another sequence combination. The table covers the two previously proposed sets of sequences ($\boldsymbol{\omega}\mathbf{m}\mathrm{p}$ and $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$) to obtain position invariance on a certain rigid body part [20], [24], applying SVD on the sequence combinations $\boldsymbol{\omega}\mathbf{ma}$, $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$, $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$, and $\boldsymbol{\omega}^{\mathrm{E}}\mathbf{m}^{\mathrm{E}}\mathrm{p}\,\delta\mathbf{q}$, and earlier proposed techniques to achieve placement (position as well as orientation) invariance on a given rigid body part [20], [24]. We processed the data on a laptop computer with a quad-core Intel Core i7-4720HQ processor at a clock speed of 2.6–3.6 GHz and 16 GB of RAM running 64-bit MATLAB R2018b.

Note that the processing time for the originally recorded sequences $\boldsymbol{\omega}\mathbf{ma}$ is null since this represents raw data, directly acquired and recorded from the sensor units. Among the data transformation techniques aimed to achieve only position invariance (Table III, rows 2–7), using the sequence combination $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$ is computationally the least costly. The SVD used for random interchangeability of the units executes faster when it processes the sequence combinations $\boldsymbol{\omega}\mathbf{m}\mathrm{p}$ or $\boldsymbol{\omega}\mathbf{ma}$ compared to $\boldsymbol{\omega}\mathbf{m}\mathrm{p}\mathbf{q}$. This is because of the calculation of the quaternion $\mathbf{q}$ in the latter, which requires estimating the sensor-unit orientation [42].

The sequence combinations in the last three rows of Table III are expressed in the Earth frame E in order to attain orientation invariance as well. Such representation again requires the estimation of the sensor-unit orientation, which is

TABLE IV
AVERAGE OF THE TOTAL RUNTIME, AVERAGE TRAINING TIME PER
CROSS-VALIDATION ITERATION, AND AVERAGE CLASSIFICATION TIME
PER TEST FEATURE VECTOR WITH ONE STANDARD DEVIATION IN
PARENTHESES

classifier	total run time (s)	training time (s)	classification time (ms)
SVM	10.68 (3.53)	9.17 (2.86)	0.42 (0.12)
ANN	6.38 (1.80)	6.37 (1.80)	0.01 (0.00)
BDM	1.41 (0.13)	0.01 (0.00)	1.37 (0.13)
LDC	1.53 (0.46)	0.27 (0.01)	0.03 (0.01)
k-NN	0.13 (0.03)	–	0.13 (0.02)
RF	26.07 (4.87)	23.23 (4.00)	0.85 (0.08)
OMP	248.20 (318.85)	–	240.87 (309.01)

responsible for most of the processing time. Applying SVD on $\boldsymbol{\omega}^E\mathbf{m}^Ep\delta\mathbf{q}$ takes longer to process than the following three because of the calculations involved in the SVD. In any case, the processing times we tabulate are all significantly less than the duration of a single time segment (5 s), indicating that the data transformations can be obtained in near real time.

Table IV displays the average runtimes of the machine learning classifiers together with one standard deviation given in parentheses over all sequence combinations considered in this article (ideally affixed units, SVD, SUBC$_1$, SUBC$_2$, and SUBC$_3$, where each of the last four is individually applied on the sequence combinations $\boldsymbol{\omega}\mathbf{ma}$, $\boldsymbol{\omega}\mathbf{m}$p, $\boldsymbol{\omega}\mathbf{m}p\mathbf{q}$, and $\boldsymbol{\omega}^E\mathbf{m}^Ep\delta\mathbf{q}$).

The second column of Table IV provides the average total runtime (comprising the training time, classification of all test feature vectors in the test phase, and programming overheads) per cross-validation iteration. It is notable that among the seven classifiers, the average total runtime of the k-NN classifier is the shortest whereas that of OMP is the longest.

The third column of Table IV displays the average training times of the classifiers per cross-validation iteration. Since the k-NN and OMP classifiers only store the training instances (feature vectors) and their class labels, their training time is zero. Contrarily, the RF classifier takes the longest to train.

The fourth column of Table IV lists the average classification time per test feature vector associated with a 5-s time segment. Even though all of the classifiers can label a test feature vector in a time interval much shorter than the associated time segment, the ANN and LDC classifiers can label almost instantly, followed by k-NN, recognizing the activity type in less than 0.13 ms. The OMP classifier, which executes an iterative algorithm independently for each test feature vector, takes the longest to classify an activity. However, its runtime is still much shorter than the duration of one segment, making near real-time execution possible.

VIII. DISCUSSION AND CONCLUSION

In this article, we have addressed the problems associated with the placement of body-worn sensor units and proposed a method that allows users to wear the sensor units interchangeably across different body parts. This is achieved through use of the principal components of the data in the classification stage, rather than the actual data itself. We have also developed a generalized classifier that recognizes activities processing the

test data acquired from only a single sensor unit that is placed at any one of the rigid body parts, while the training data had to be acquired from multiple units on different body parts. We have merged and extended both approaches with our former methodology for achieving position invariance on the same rigid body part.

We have further extended our methodology to attain invariance to both the position and orientation of body-worn motion-sensor units simultaneously. This permits the user to wear the sensor units freely at any position and orientation on a given rigid body part. This is achieved with the proviso that data were collected from that body part with ideal sensor-unit placement and the wearable system was trained with those data. In the case of the generalized classifier, it is sufficient to affix only one sensor unit at any position and orientation on any one of the rigid body parts from which training data were collected. More importantly, the proposed approaches substantially reduce the amount of data/time required for training since it is no longer necessary to train the system for every possible placement configuration (position and orientation) of each sensor unit. Overall, we have observed that there is a tradeoff to be made between the flexibility allowed to the user in sensor-unit placement and activity classification rate.

We have assessed the performances of the proposed approaches using our publicly available data set containing two major activity categories where the activities are larger in number and more complicated than those considered in the existing work. We have implemented seven state-of-the-art machine learning classifiers and two cross-validation techniques to demonstrate the performance of our methodology.

The accuracy values appear to be low for some of the considered approaches. There are multiple factors that contribute to this. First, some of the activities in the data set are likely to be confused with each other especially when sensor units are placed differently. For instance, it is difficult, if not impossible, to distinguish the four stationary activities (sitting, standing, supine, and reclined-right side) from each other when there is limited or no information about the placement of the sensor unit on the body. Second, there are activities that are inherently similar to each other. For instance, walking on a flat and 15°-inclined treadmill exhibit similar body movements. Third, there is naturally significant variability in the activities performed by different subjects, which is the main reason for the dissimilarities observed in P-fold and L1O cross-validation results. When we allow flexibility in the placement configuration of the sensor units, we essentially remove the placement-related cues from the acquired data, which makes it more challenging to handle the variations among subjects within the activities. This is why the proposed methods perform better for P-fold than for L1O, where the training and test sets in the latter are associated with different subjects.

The methodology described in this article does not use information on the activity types and the positions of the sensor units because our aim is to propose sufficiently general techniques, applicable to a broad range of wearable systems and scenarios. The proposed sequence combinations were derived from each data segment independently. This way,

the effect of a sudden change or shift in the placement of the sensor units was kept confined to the time segment during which the change occurs and will not affect the system performance in the subsequent time segments. The methodology employs multidimensional time-domain sequences with a structure similar to that of the originally acquired data, which comprises nine time-domain sequences per sensor unit. Thus, it is straightforward to integrate it into a wide variety of wearables by preprocessing the sensory data without considerable effort. Consequently, the system can tolerate variability in sensor-unit placement and its performance is relatively robust to deviations in the positions and orientations of the sensor units on the body.

It is possible to generalize our proposed methods to position and orientation shifts *within* a rigid body part as well (rather than being confined to random shifts on its *surface*). In [20], although we validated our proposed position-invariance techniques by simulating randomly displaced sensor units on the surfaces of rigid body parts, the techniques do not exploit this information and are expected to be generalizable in this respect. However, wearable sensor units are typically displaced and misoriented *on* the surface of the human body (on the skin or clothing) unless they are implanted.

As future work, additional techniques to gain invariance to the placement configuration of the sensor units on the body can be investigated. Bilateral symmetry of the human body can be exploited and differential signals between sensor units can be considered. The study can be extended to achieve invariance to placement configuration of other elements of IoT; e.g., sensors embedded in smart environments as well as implanted sensors.

REFERENCES

[1] V. Bianchi, M. Bassoli, G. Lombardo, P. Fornacciari, M. Mordonini, and I. de Munari, "IoT wearable sensor and deep learning: An integrated approach for personalized human activity recognition in a smart home environment," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8553–8562, Oct. 2019.

[2] I. Bisio, C. Garibotto, A. Grattarola, F. Lavagetto, and A. Sciarrone, "Exploiting context-aware capabilities over the Internet of Things for Industry 4.0 applications," *IEEE Netw.*, vol. 32, no. 3, pp. 101–107, May/Jun. 2018.

[3] L. Chen and C. D. Nugent, *Human Activity Recognition and Behaviour Analysis for Cyber-Physical Systems in Smart Environments*. Cham, Switzerland: Springer, 2019.

[4] A. Patel and J. Shah, "Sensor-based activity recognition in the context of ambient assisted living systems: A review," *J. Ambient Intell. Smart Environ.*, vol. 11, no. 4, pp. 301–322, Jul. 2019.

[5] A. Yang, C. Zhang, Y. Chen, Y. Zhuansun, and H. Liu, "Security and privacy of smart home systems based on the Internet of Things and stereo matching algorithms," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2521–2530, Apr. 2020.

[6] Y. Wang, S. Cang, and H. Yu, "A survey on wearable sensor modality centred human activity recognition in health care," *Expert Syst. Appl.*, vol. 137, pp. 167–190, Dec. 2019.

[7] M. Keally, G. Zhou, G. Xing, J. Wu, and A. J. Pyles, "PBN: Towards practical activity recognition using smartphone-based body sensor networks," in *Proc. 9th ACM Conf. Embedded Netw. Sensor Syst. (SenSys)*, Seattle, WA, USA, Nov. 2011, pp. 246–259.

[8] I. Bisio *et al.*, "Towards IoT-based eHealth services: A smart prototype system for home rehabilitation," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019.

[9] K. Altun, B. Barshan, and O. Tunçel, "Comparative study on classifying human activities with miniature inertial and magnetic sensors," *Pattern Recognit.*, vol. 43, no. 10, pp. 3605–3620, Oct. 2010.

[10] O. Tunçel, K. Altun, and B. Barshan, "Classifying human leg motions with uniaxial piezoelectric gyroscopes," *Sensors*, vol. 9, no. 11, pp. 8508–8546, Nov. 2009.

[11] B. Barshan and M. C. Yüksek, "Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units," *Comput. J.*, vol. 57, no. 11, pp. 1649–1667, Nov. 2014.

[12] L. Gutiérrez-Madroñal, L. La Blunda, M. F. Wagner, and I. Medina-Bulo, "Test event generation for a fall-detection IoT system," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6642–6651, Aug. 2019.

[13] A. T. Özdemir and B. Barshan, "Detecting falls with wearable sensors using machine learning techniques," *Sensors*, vol. 14, no. 6, pp. 10691–10708, Jun. 2014.

[14] A. Yurtman and B. Barshan, "Automated evaluation of physical therapy exercises using multi-template dynamic time warping on wearable sensor signals," *Comput. Methods Programs Biomed.*, vol. 117, no. 2, pp. 189–207, Nov. 2014.

[15] J. Morales and D. Akopian, "Physical activity recognition by smartphones, a survey," *Biocybern. Biomed. Eng.*, vol. 37, no. 3, pp. 388–400, 2017.

[16] M. Shoaib, S. Bosch, Ö. D. İncel, H. Scholten, and P. J. M. Havinga, "A survey of online activity recognition using mobile phones," *Sensors*, vol. 15, no. 1, pp. 2059–2085, Jan. 2015.

[17] I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone, "Smartphone-based user activity recognition method for health remote monitoring applications," in *Proc. 2nd Int. Conf. Pervasive Embedded Comput. Commun. Syst. (PECCS)*, Rome, Italy, Feb. 2012, pp. 200–205.

[18] M. S. Siraj *et al.*, "UPIC: User and position independent classical approach for locomotion and transportation modes recognition," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput. ACM Int. Symp. Wearable Comput. (UbiComp/ISWC Adjunct)*, Virtual Event, Mexico, Sep. 2020, pp. 340–345.

[19] K. Kunze and P. Lukowicz, "Sensor placement variations in wearable activity recognition," *IEEE Pervasive Comput.*, vol. 13, no. 4, pp. 32–41, Oct.–Dec. 2014.

[20] B. Barshan and A. Yurtman, "Classifying daily and sports activities invariantly to the positioning of wearable motion sensor units," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 4801–4815, Jun. 2020.

[21] A. Yurtman and B. Barshan, "Activity recognition invariant to sensor orientation with wearable motion sensors," *Sensors*, vol. 17, no. 8, Aug. 2017, Art. no. 1838.

[22] A. Yurtman, B. Barshan, and B. Fidan, "Activity recognition invariant to wearable sensor unit orientation using differential rotational transformations represented by quaternions," *Sensors*, vol. 18, no. 8, Aug. 2018, Art. no. 2725.

[23] K. Altun and B. Barshan, *Daily and Sports Activities Dataset*, IEEE Data Port, Feb. 2019, doi: 10.21227/at1v-6f84.

[24] A. Yurtman, "Activity recognition invariant to position and orientation of wearable motion sensor units," Ph.D. dissertation, Dept. Elect. Electron. Eng., Bilkent Univ., Ankara, Turkey, Apr. 2019.

[25] T. Hur, J. Bang, D. Kim, O. Banos, and S. Lee, "Smartphone location-independent physical activity recognition based on transportation natural vibration analysis," *Sensors*, vol. 17, no. 4, Apr. 2017, Art. no. 931.

[26] M. Jiang, H. Shang, Z. Wang, H. Li, and Y. Wang, "A method to deal with installation errors of wearable accelerometers for human activity recognition," *Physiol. Meas.*, vol. 32, no. 3, pp. 347–358, Feb. 2011.

[27] D. Ravì, C. Wong, B. Lo, and G.-Z. Yang, "A deep learning approach to on-node sensor data analytics for mobile or wearable devices," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 56–64, Jan. 2017.

[28] A. M. Khan, M. H. Siddiqi, and S.-W. Lee, "Exploratory data analysis of acceleration signals to select light-weight and accurate features for real-time activity recognition on smartphones," *Sensors*, vol. 13, no. 10, pp. 13099–13122, Sep. 2013.

[29] A. M. Khan, A. Tufail, A. M. Khattak, and T. H. Laine, "Activity recognition on smartphones via sensor-fusion and KDA-based SVMs," *Int. J. Distrib. Sens. Netw.*, vol. 10, no. 5, May 2014, Art. no. 503291.

[30] O. Banos, M. A. Toth, M. Damas, H. Pomares, and I. Rojas, "Dealing with the effects of sensor displacement in wearable activity recognition," *Sensors*, vol. 14, no. 6, pp. 9995–10023, Jun. 2014.

[31] R. Chavarriaga, H. Bayati, and J. del R. Millán, "Unsupervised adaptation for acceleration-based activity recognition: Robustness to sensor displacement and rotation," *Pers. Ubiquitous Comput.*, vol. 17, no. 3, pp. 479–490, Mar. 2013.

[32] K. Förster, D. Roggen, and G. Tröster, "Unsupervised classifier self-calibration through repeated context occurrences: Is there robustness against sensor displacement to gain?" in *Proc. Int. Symp. Wearable Comput.*, Linz, Austria, Sep. 2009, pp. 77–84.

[33] K. Kunze, P. Lukowicz, H. Junker, and G. Tröster, "Where am I: Recognizing on-body positions of wearable sensors," *Location- and Context-Awareness* (LNCS 3479), T. Strang and C. Linnhoff-Popien, Eds. Heidelberg, Germany: Springer, 2005, pp. 264–275.

[34] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell, "The Jigsaw continuous sensing engine for mobile phone applications," in *Proc. 8th ACM Conf. Embedded Netw. Sensor Syst.*, Zürich, Switzerland, Nov. 2010, pp. 71–84.

[35] H. Martín, A. M. Bernardos, J. Iglesias, and J. R. Casar, "Activity logging using lightweight classification techniques in mobile devices," *Pers. Ubiquitous Comput.*, vol. 17, no. 4, pp. 675–695, Apr. 2013.

[36] T. Sztyler and H. Stuckenschmidt, "On-body localization of wearable devices: An investigation of position-aware activity recognition," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Sydney, NSW, Australia, Mar. 2016.

[37] J. Wu and R. Jafari, "Orientation independent activity/gesture recognition using wearable motion sensors," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1427–1437, Apr. 2019.

[38] K. Förster, P. Brem, D. Roggen, and G. Tröster, "Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks," in *Proc. 5th Int. Conf. Intell. Sensors Sensor Netw. Inf. Process. (ISSNIP)*, Melbourne, VIC, Australia, Dec. 2009, pp. 43–48.

[39] W. Xu, M. Zhang, A. A. Sawchuk, and M. Sarrafzadeh, "Robust human activity and sensor location corecognition via sparse signal representation," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 11, pp. 3169–3176, Nov. 2012.

[40] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Using mobile phones to determine transportation modes," *ACM Trans. Sensor Netw.*, vol. 6, no. 2, Art. no. 13, Feb. 2010.

[41] J. L. Meriam, L. G. Kraige, and J. N. Bolton, *Engineering Mechanics: Dynamics*, 9th ed. New York, NY, USA: Wiley, Mar. 2018.

[42] A. Yurtman and B. Barshan, "Novel non-iterative orientation estimation for wearable motion sensor units acquiring accelerometer, gyroscope, and magnetometer measurements," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 3206–3215, Jun. 2020.

[43] J. B. Kuipers, *Quaternions and Rotation Sequences: A Primer With Applications to Orbits, Aerospace, and Virtual Reality*. Princeton, NJ, USA: Princeton Univ. Press, 1999.

[44] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Baltimore, MD, USA: John Hopkins Univ. Press, 2013.

[45] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," Dept. Aeronaut. Astronaut., Stanford Univ., Stanford, CA, USA, Rep., Oct. 2006. [Online]. Available: http://www.swarthmore.edu/NatSci/mzucker1/papers/diebel2006attitude.pdf

[46] B. Barshan and A. Yurtman, "Investigating inter-subject and inter-activity variations in activity recognition using wearable motion sensors," *Comput. J.*, vol. 59, no. 9, pp. 1345–1362, Sep. 2016.

[47] K. Altun and B. Barshan, *Daily and Sports Activities Dataset*, UCI Mach. Learn. Repository, School Inf. Comput. Sci., Univ. California at Irvine, Irvine, CA, USA, 2013. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Daily+and+Sports+Activities

[48] *MTi, MTx, and XM-B User Manual and Technical Documentation*, Xsens Technol. B.V., Enschede, The Netherlands, 2021. [Online]. Available: http://www.xsens.com

[49] A. R. Webb, *Statistical Pattern Recognition*, 2nd ed. West Sussex, U.K.: Wiley, 2002.

[50] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," Dept. Comput. Sci., Nat. Taiwan Univ., Taipei, Taiwan, Tech. Rep., 2003.

[51] K.-B. Duan and S. S. Keerthi, "Which is the best multiclass SVM method? An empirical study," *Multiple Classifier Systems* (LNCS 3541), N. C. Oza, R. Polikar, J. Kittler, and F. Roli, Eds. Heidelberg, Germany: Springer, 2005, pp. 278–285.

[52] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, Art. no. 27, Apr. 2011. [Online]. Available: http://www.csie.ntu.edu.tw/ cjlin/libsvm/

[53] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, May 2007.

[54] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2000.

[55] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Cambridge, MA, USA: Elsevier, 2016.

[56] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, USA, Nov. 1993, pp. 40–44.

[57] L. Wang, L. Cheng, and G. Zhao, *Machine Learning for Human Motion Analysis: Theory and Practice*. Hershey, PA, USA: IGI Global, 2010.

Aras Yurtman received the B.S., M.S., and Ph.D. degrees in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2010, 2012, and 2019, respectively.

He is currently a Postdoctoral Researcher with the Declarative Languages and Artificial Intelligence Research Group, Department of Computer Science, Katholieke Universiteit at Leuven, Leuven, Belgium. He was a Research and Teaching Assistant and an Administrative Assistant with Bilkent University. His current research interests include active learning, semisupervised clustering, explainable machine learning, and time-series analysis.

Billur Barshan received the B.S. degrees in electrical engineering and in physics from Boğaziçi University, Istanbul, Turkey, and the M.S., M.Phil., and Ph.D. degrees in electrical engineering from Yale University, New Haven, CT, USA, in 1986, 1988, 1988, and 1991, respectively.

After working as a Postdoctoral Researcher with the Robotics Research Group, University of Oxford, Oxford, U.K., she joined the Faculty of Bilkent University, Ankara, Turkey, where she is currently a Professor with the Department of Electrical and Electronics Engineering. Her current research interests include wearable sensing, wearable robots and mechanisms, intelligent sensing, motion capture and analysis, detection and classification of falls, machine learning, pattern classification, and multisensor data fusion.

Prof. Barshan received the TÜBİTAK Young Scientist Award in 1998, the METU Mustafa Parlar Foundation Research Award in 1999, and two best paper awards. She served on the Management Committee of the COST-IC0903 Action MOVE from 2010 to 2013.

Soydan Redif (Senior Member, IEEE) received the B.Eng. degree (First Class Hons.) in electronic engineering from Middlesex University, London, U.K., and the Ph.D. degree in electrical and electronic engineering from the University of Southampton, Southampton, U.K., in 1998 and 2006, respectively.

From 1998 to 2007, he was a Research Scientist with DERA/QinetiQ, Malvern, U.K., joining the Malvern Signal Processing Group in 2000. Since 2008, he has held academic appointments in North Cyprus. He has been with the European University of Lefke, Lefke, North Cyprus, since 2011, where he is currently an Associate Professor with the Department of Electrical and Electronics Engineering. His current research interests include adaptive and array signal processing applied to source separation, communications, biomedical, and wearable systems.

Dr. Redif received the IEE Award for Outstanding Academic Achievement in 1998. He was a Co-Organizer and the Session Chair for ICASSP 2019, Brighton, U.K. He is a U.K. Chartered Engineer and a member of IET.