Neural Processing Letters
https://doi.org/10.1007/s11063-021-10468-z

Opportunistic Activity Recognition in IoT Sensor Ecosystems via Multimodal Transfer Learning

Oresti Banos[1] · Alberto Calatroni[2] · Miguel Damas[1] · Hector Pomares[1] · Daniel Roggen[3] · Ignacio Rojas[1] · Claudia Villalonga[4]

Accepted: 15 February 2021

Abstract
Recognizing human activities seamlessly and ubiquitously is now closer than ever given the myriad of sensors readily deployed on and around users. However, the training of recognition systems continues to be both time and resource-consuming, as datasets must be collected ad-hoc for each specific sensor setup a person may encounter in their daily life. This work presents an alternate approach based on transfer learning to opportunistically train new unseen or target sensor systems from existing or source sensor systems. The approach uses system identification techniques to learn a mapping function that automatically translates the signals from the source sensor domain to the target sensor domain, and vice versa. This can be done for sensor signals of the same or cross modality. Two transfer models are proposed to translate recognition systems based on either activity templates or activity models, depending on the characteristics of both source and target sensor systems. The proposed transfer methods are evaluated in a human–computer interaction scenario, where the transfer is performed in between wearable sensors placed at different body locations, and in between wearable sensors and an ambient depth camera sensor. Results show that a good transfer is possible with just a few seconds of data, irrespective of the direction of the transfer and for similar and cross sensor modalities.

Keywords Transfer learning · Multimodal sensors · Wearable sensors · Ambient sensors · Activity recognition · Human–computer Interaction

✉ Oresti Banos
oresti@ugr.es

1 Department of Computer Architecture and Computer Technology, Research Center for Information and Communication Technologies of the University of Granada (CITIC-UGR), C/Periodista Rafael Gmez Montero 2, 18071 Granada, Spain

2 Bonsai Systems GmbH, Kraehbuehlstrasse 58, 8044 Zurich, Switzerland

3 Sensor Technology Research Centre, University of Sussex, Falmer, Brighton BN1 9QT, United Kingdom

4 School of Engineering and Technology, Universidad Internacional de la Rioja, Av. de la Paz 137, 26006 Logroño, Spain

Published online: 31 March 2021

1 Introduction

The Internet-of-Things (IoT) is revolutionizing the world as we know it. IoT is virtually everywhere, supporting in novel ways a myriad of application areas ranging from transportation and logistics to industry or healthcare, among others [1]. One such application area is the automatic recognition of human activities [2], which has recently attracted considerable attention. Activity recognition systems are typically designed around a limited set of predefined sensors that are selected to be highly discriminative of the activities of interest [3–5]. The far-ranging spectrum of sensors and devices available nowadays to users, along with the unprecedented connectivity and computation resources offered by the IoT, enable a new era where activity recognition may be realized continuously and opportunistically.

IoT sensor ecosystems, i.e. connected body and ambient sensor networks, are however subject to dynamic changes posing important challenges from a sensing [6] and processing [7] perspective. Such changes may affect the normal operation of activity recognition systems. For example, sensors can suffer from irrecoverable failures that demand a replacement of the affected sensor by a new one of similar or close characteristics. Sensors can also be replaced when higher efficiency or new features are sought, for example, to reduce power consumption. Most often, users acquire new sensor devices to benefit from other services not supported by the current sensor ecosystem. For all these cases, the newcomer sensor may potentially have different characteristics to the substituted one (e.g. different sampling rate, dynamic range, modality). This leads to a new sensor setup hardly foreseeable during the design of the activity recognition system. As a result, default activity recognition systems may not directly operate on the data obtained through the newly encountered sensor ecosystem.

Changes in the IoT sensor ecosystem do not only occur upon maintenance or upgrades. In fact, different sensor configurations are envisioned during a user's normal day [8]. Depending on the particular context, users may, for example, wear smart accessories (e.g. at the gym), interact with video gaming systems (e.g. at home), or use instrumented tools (e.g. at work). There is a clear tendency towards increased availability of sensors readily deployed by users by themselves (e.g. smartphones, sensor-equipped gadgets, smart objects, smart clothing) or integrated as part of living environments (e.g. sensors for climate control, security, entertainment). Overall, as users encounter different situations and contexts, sensors may be removed, substituted, or newly added [9], thus providing different sensing "opportunities". In the general case, many of these sensors may not have associated activity models to use them for activity recognition, as they are deployed for other purposes. However, most of this sensing equipment could be used for activity recognition since they may be in principle capable of measuring human behavior (e.g. body motion).

In case the replaced sensor is different from the predecessor one, or a new sensor is introduced in the network, a full retraining of the activity recognition system is required. Likewise, a new model must be built if recognition capabilities are to be given to a sensor originally not devised for activity recognition tasks. Following a conventional learning process is in both cases quite costly since this normally requires to collect new experimental data. Such data collection implies to record the behavior of a person or group of people while performing the activities of interest for the new sensor setup. Besides being a long and tedious process, sensor setups may vary from person to person, or even from one context to another, thus this approach proves impractical for real-world applications.

The training of newcomer sensors should ideally be performed without intervention of a system designer, which would otherwise limit the approach to predefined sensor setups and deployments. This must also happen without user involvement. To meet these requirements,

the most reasonable approach is to use the actual knowledge of the existing activity recognition system to instruct the new sensors on the activity recognition tasks. This is accomplished in a process in which the original activity recognition system or "teacher" transfers its knowledge to the newcomer untrained sensor or "learner". Also known as transfer learning [10], this corresponds to a research problem in machine learning that focuses on translating the knowledge available to solve a problem in one domain to a different but related domain.

This work presents an automatic approach to translate activity recognition systems between sensor domains, thereby effectively enabling to transfer activity recognition capabilities from an existing or source system to an untrained or target system. The approach relies on the learning of a mapping between source and target sensor signals through the use of system identification techniques. Two transfer modes are proposed for the translation of activity recognition systems that operate either on activity templates, i.e. signal patterns representing activities or gestures, or activity models, i.e. trained machine learning models. The choice of transfer mode depends on the characteristics of the source and target systems. Both mapping and transfer are performed in a short time and for similar or cross modalities, thus supporting continuity of recognition and the opportunistic use of sensors as they become available to the user.

The rest of the paper is organised as follows. Section 2 presents related works using transfer learning for activity recognition purposes. Section 3 describes the system identification and transfer learning methodology proposed. The performance of the transfer methods is evaluated for various sensor domains and modalities in Sect. 4. Results are discussed in Sect. 5, while the conclusions are summarized in Sect. 6.

2 Related Work

Different approaches have been proposed in previous works to increase the tolerance to changes in the sensing environment. The simplest yet least effective approach to cope with such variations consists in collecting data for a handful of sensor setups intended to represent the environment encountered at run-time. Thus, for example, sensor-placement-independent activity recognition can in principle be achieved by using datasets collected from multiple on-body locations [11]. Unfortunately, this approach is highly inefficient as it requires to record data for an enormous, ideally infinite number of sensor placement combinations, thus leading to great data collection efforts and a highly time-demanding user participation. Self-calibration approaches can be a good alternative as they do not require users' intervention. However, these approaches have been demonstrated fairly applicable for just a few cases involving sensor displacement [12,13]. Combinations of multiple sensor modalities can also help to tolerate sensor displacement [14] or to substitute sensor modalities [15]. Nevertheless, these combinations must be predefined for some selected types of sensor variations. Alternatively, sensors can self-characterize their placement [16] and orientation [17] to select the appropriate activity models at run-time. Such models must be also predefined in any case. Advanced geometrical transformations of the sensor data streams have been shown to support some invariance to changes in the orientation of sensors [18,19]. The use of fusion and ensemble techniques have also been proposed to cope with variations in the infrastructure due to sensor failures [20] and sensor displacement [21,22]. One main limiting factor of most of these approaches is that they are generally constrained to foreseen run-time variations. Moreover, either activity models or ad-hoc transformations between modalities must be designed for each case.

Transfer learning is devised here as a means to overcome the aforementioned limitations, making it particularly suitable to support run-time variations in the sensor setup. From a taxonomical point of view, three levels of transfer learning have mainly been explored in activity recognition. The lowest level corresponds to the transfer of activity instances or templates, where all or some of the samples in the source domain are reused in the learning of the target recognition task to compensate for the low number of target training samples [23]. The intermediate level addresses the transfer of feature representations or feature spaces, where the focus is on learning a good knowledge representation model for the target domain based on relevant information from the source domain [24]. The highest level is the transfer of activity models or parameters, where models trained in the source domain are used to regularize or to be used in the model learning of the target recognition task [25]. More recently, the interdependence between knowledge representation and transfer type in activity recognition has also been studied [26]. The authors explore various high-level semantic knowledge representations on different transfer types through a generic transfer metric learning framework. Furthermore, they introduce a hierarchical knowledge representation model based on the embedded structure in the semantic attribute space.

Diverse transfer learning-based solutions have been proposed in the activity recognition domain to handle unforeseen sensor setup variations at execution time. Some approaches leverage existing labelled training data in the source domain to interpret the equivalent data in the target domain by mining the similarities between the activities in the two domains [27,28]. Other works define profiles for sensors in the source and target domain using background knowledge about the sensor networks, which is later used to measure the similarity of features extracted from different datasets [29]. Matching algorithms are then adopted to automatically compute appropriate mappings of features based on the similarity measure. There exist several approaches exploiting the use of heuristic searches at the feature level to procure the transfer between domains with different feature spaces [30–33]. This is normally accomplished without requiring typical feature–feature, feature–instance, or instance–instance co-occurrence data. Instead, the features are related in different feature spaces through the construction of metafeatures. The power of deep convolutional networks as feature extractors has also been leveraged to procure transfer of activity recognition capabilities [34]. The authors propose reusing kernels learned on a source domain on another target domain, thus supporting transfer between users, application domains, sensor modalities and sensor locations. Automatic real-time retraining of pretrained machine learning algorithms has also been proposed as a means to transfer knowledge [35]. In this case, the inherent correlation between observations made by an old sensor, for which trained algorithms exist, and the new sensor is explored.

Practically, principles allowing a trained system to transfer activity recognition capabilities to another system were already proposed in [36,37]. In these works, an initial system recognizes activities and supervises the learning of a new one, without user specific intervention. Such approaches work across sensor modalities and are characterized on different body-worn and ambient sensors. However, they require to operate on long time scales as they need all the relevant activities to be observed several times, e.g. timescale of days or more. Besides, these methods are prone to incomplete transfer learning since it is likely that the user does not perform the complete set of target activities.

3 Multimodal Transfer Methods

Overall, existing approaches do not fulfil the characteristics desired in this work since they either need to predefine allowed run-time variations, cannot operate on short time scales, or are not defined for adaptation across sensor modalities. Through the use of transfer learning concepts, this work aims at making it possible to apply the activity templates (signal patterns representing activities) or models (trained machine learning model) of a given sensor system (source domain) to the data from a newly discovered sensor system (target domain) which lacks such templates or models. It is assumed that source and target sensor systems coexist for a short time and that there is an unknown function, to be determined, that allows for mapping between their signals. Besides, these signals can be of similar or different modality. In the following, the models proposed to identify these relations as well as to transfer the activity recognition templates and models are presented.

The proposed transfer methods work in two steps. First, a system identification technique (Sect. 3.1) finds a function that maps the signals of the source sensor to the signals of the target sensor. This process can be carried out on signals corresponding to the same modality (identical transfer) or different modality (cross transfer). Based on this mapping, the activity recognition system is then transferred. For the activity recognition system based on templates (Sect. 3.2), the transfer process consists in translating the activity templates from the source to the target domain. These new templates may be directly used for recognition or to build an activity model for the target sensor. For the recognition system based on activity models (Sect. 3.3), the transfer process consists in literally conveying or copying the source activity model to the target system. To operate on this activity model the target system needs to map its signals to the source domain in which the activity model was originally defined.

3.1 System Identification

The complexity of the signal mapping transfer stems from the physics of the domain transformation. In some cases the underlying relationship is well-defined and known (e.g. position to acceleration) but for other cases it may not be that clearly identifiable (e.g. position to magnetic field). In addition, the setup constraints associated to each particular context and subject reduce the generalization capabilities of those models which are problem-specific. In many cases such models will be overly convoluted and almost impossible to obtain in reasonable time due to the complex nature of most systems and processes. Furthermore, the intricacy of this engineering design increases dramatically with the number of sensors, thus constituting a non-scalable approach.

The transformation or mapping of data from one domain to a similar or different domain is normally approached in base of the a priori knowledge of the underlying relationship among domains. White-box (WB) models, also known as user-defined models, are typically preferred when all the necessary information about the problem domain is available. Hence, WB designs require to know for instance the placement and nature of the sensors, the features delivered by each sensor node (e.g. data range, units) or the number and type of sensors (modalities) to define an appropriate mapping transfer function. This task is extremely time and resource-consuming because a design team is required to analyze and implement the models for each combination of sensors and domains. Moreover, this approach does not fulfill the desired autonomous characteristic of the system, where no prior knowledge of the problem domain and sensor ecosystem must be assumed.

Black-box (BB) models are encountered to be more appropriate for this problem as they are data-driven models where only the inputs (source) and outputs (target) are needed, with limited additional knowledge about the internals of the model [38]. BB modeling is often used when assumptions on the nature of the underlying system are hard to make, when the complexity of the underlying relation is extremely high, or to avoid designer's bias, which fits in well with the characteristics of this problem. The choice of the mathematical functions within BB models is normally made depending on whether the system to model is linear or non-linear, and whether it is time-dependent or time-independent. A system can also have single or multiple inputs and/or outputs or explicitly incorporate external exogenous disturbances such as noise. In summary, the most interesting characteristics of the use of BB models instead of a classical WB approach are:

- *Generalization capabilities*: it can be in principle used for whichever kind of mapping, even combining different type of domain sources or using abstract magnitudes, such as proximity or object-interaction sensors (e.g. switch buttons, RFID).
- *Scalability*: in general the model may be applied with no much effort to a larger set of sensors or signals, i.e. inputs, of different nature.
- *Robustness to information loss*: the statistical capabilities of some of its regression and prediction models allow for learning even when data loss events are present.
- *Design complexity*: there is no need to explicitly discover the underlying relation that links the domain transformation; no extra information about the sensor deployment is a priori required; reduced design time, normally automatic and autonomously performed.
- *User abstraction*: no specific user intervention is required, thus avoiding burdensome data collection procedures for the training of the systems.

The system identification model should allow for transformations between the regular sensing modalities that are used for activity recognition, most of which are of a linear nature. Some typical static transformations include scaling (sensors with different sensitivity or units, whether absolute or relative) and affine transformations, offset (mean different to zero), non-linearity (compression of the dynamic range), and translation or rotation (e.g. when an acceleration sensor is displaced, translated and/or rotated). Dynamic transformations may include multiple differentiation or integration operations (e.g. between position or angle and linear or angular velocity), or hysteresis. Besides, most sensors used in activity recognition problems measure various axes at once (e.g. triaxial accelerometers, 3D positioning systems) which are not independent, so the system identification model has to be of the multiple-input-multiple-output (MIMO) kind.

Accordingly, this work proposes the use of a parametric linear model, which surpasses non-linear approaches in terms of [39]: interpretation (represents and extracts the properties and knowledge of the underlying relationship); generalization (captures the true dynamics and predicts accurately the output for unseen new data); robustness to overfitting and noise rejection; speed; and amount of data required for the training and complexity (training time, computational resources required, etc.). For its computational simplicity and to procure a high velocity of mapping discovery, a linear MIMO mapping is specifically used for system identification [40]. Such mappings can be directly learned from data. This approach enables to learn mappings in a wide range of sensing environments without designer involvement or bias. In the following, the mathematical description of the considered models is provided.

Let us define $\mathbf{x_S}(t)$ as an n_S-by-1 vector of sensor data from the source domain S at time t and $\mathbf{x_T}(t)$ as an n_T-by-1 vector of data of the sensors of the target domain T. A mapping relating the sensor signals in different domains is first identified. This may be from source to target signals, or target to source signals, whichever can be best identified. Let us denote with

$\Psi_{S\to T}$ the function that maps the source to the target signal: $\Psi_{S\to T} : \mathbf{x_S}(t) \to \hat{\mathbf{x}}_\mathbf{T}(t) \approx \mathbf{x_T}(t)$. $\Psi_{T\to S}$ defines as the function that maps the target to the source signal: $\Psi_{T\to S} : \mathbf{x_T}(t) \to \hat{\mathbf{x}}_\mathbf{S}(t) \approx \mathbf{x_S}(t)$. The $\hat{}$ symbol is used to indicate that the signal is predicted in a given domain from the known signal of another domain.

A linear MIMO mapping is defined as follows:

$$\mathbf{x_T}(t) = \mathbf{B(l)}\mathbf{x_S}(t) \tag{1}$$

where $\mathbf{B(l)}$ is a n_T-by-n_S polynomial matrix in the delay operator l^{-1} (each entry of the matrix is a polynomial in l^{-1}). The operator l^{-k} introduces a delay of k samples in the signal to which it is applied: $l^{-k}x(t) = x(t-k)$. The source and target sensor signals are the inputs and outputs of the model. The matrix $\mathbf{B(l)}$ contains elements $b_{ik}(l)$,

$$\mathbf{B(l)} = \begin{pmatrix} b_{11}(l) & b_{12}(l) & \cdots & b_{1n_S}(l) \\ b_{21}(l) & b_{22}(l) & \cdots & b_{2n_S}(l) \\ \vdots & \vdots & \ddots & \vdots \\ b_{n_T1}(l) & b_{n_T2}(l) & \cdots & b_{n_Tn_S}(l) \end{pmatrix} \tag{2}$$

of the form:

$$b_{ik}(l) = b_{ik}^{(0)} l^{-s_{ik}} + b_{ik}^{(1)} l^{-s_{ik}-1} + \cdots + b_{ik}^{(q)} l^{-s_{ik}-q} \tag{3}$$

where q is the number of past input samples that are used for the computation of the current output sample and s_{ik} are the static delays from the k-th input to the i-th output. Otherwise, $b_{ik}(l)$ represents the transfer function in the Z-transform domain from the k-th input (k-th channel of the source system) to the i-th output (i-th channel of the target system). In this way, $\mathbf{B(l)}$ accounts for the contributions of all inputs to calculate the outputs. For the identification of the $(q+1) \times n_T \times n_S$ coefficients of the polynomials and the $n_T \times n_S$ static delays, a least squares approach is followed. QR factorization solves the overdetermined set of linear equations that constitutes the least-squares estimation problem. Internal loop feedback is not considered here for the sake of simplicity, thus the transfer function is rather devised as a forward combination of the inputs, i.e. a linear combination of the tapped delay inputs.

The linear MIMO mapping allows for combinations of subsets of the transformations mentioned above:

- Scaling. This is obtained by setting $b_{ik}^{(0)}$ to the scaling factor and $b_{ik}^{(s)}$ to zero $\forall s > 0, \forall i = k$. Furthermore, all the coefficients $b_{ik}^{(s)}, i \neq k$ of the off-diagonal polynomials will be zero, yielding a diagonal matrix.
- Rotation. This is obtained by setting $b_{ik}^{(0)}$ to the corresponding element at position ik in the rotation matrix and by setting $b_{ik}^{(s)}$ to zero $\forall\, s > 0$.
- Differentiation of order h. This is obtained by setting $b_{ik}^{(s)}, \forall\, s \leq h, \forall\, i = k$ to the corresponding coefficients of the transfer function of the derivative. All the other coefficients are set to zero.

3.2 Transfer of Activity Templates

Given a source activity recognition system defined through activity templates T_S, i.e. signal patterns that represent certain activities or gestures, the aim is to transfer this system recognition capabilities to a new target system. To that end, the source system activity templates need to be translated to the target system domain, where the latter may use them for recognition tasks (namely for signal recognition or pattern matching, e.g. dynamic time warping

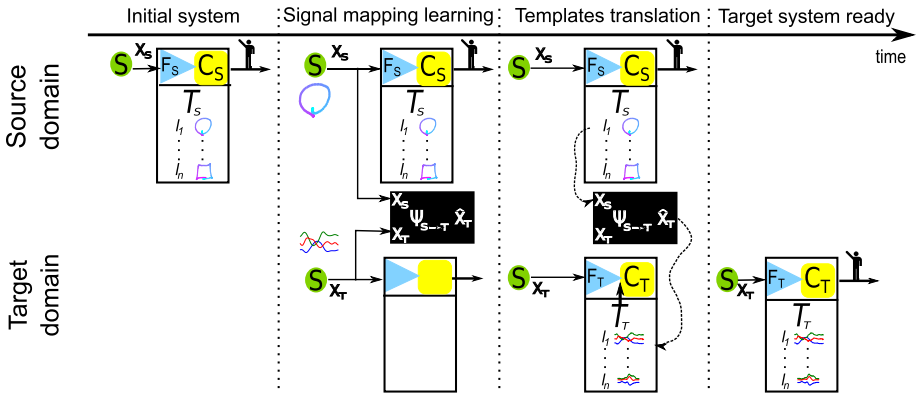


Fig. 1 Architecture for the transfer of activity templates. From left to right, initially a fully operational activity recognition system is identified, hereafter the source system. Such system is defined through a number of activity templates T_S, which are normally transformed into features F_S and used to train a classifier C_S. Then, a mapping function $\Psi_{S\rightarrow T}$ between source and target domains is discovered through system identification. Thereafter, the activity templates T_S are translated from source to target domain, thus allowing the target system to use the translated templates T_T to build its own activity recognition system (F_T, C_T). Finally, the target system is ready to operate. Note: the depicted signals may for example represent position (source domain) and acceleration (target domain)

[41], shapelets [42], etc., or for building more sophisticated activity models through feature extraction and classification procedures).

The complete architecture of the method is depicted in Fig. 1. It starts from a source operational activity recognition system that recognizes activities from the data of a sensor. The recognition system devised for the source domain also stores the activity templates T_S. T_S consists of raw sensor signals $\mathbf{x_S}(t)$ and the corresponding class labels. First, a mapping function $\Psi_{S\rightarrow T}$ between source and target sensor signals is obtained through system identification. Then, $\Psi_{S\rightarrow T}$ is used to translate the templates T_S into templates T_T containing the predicted sensor signals $\hat{\mathbf{x}}_\mathbf{T}(t)$ in the target domain, and the corresponding class labels. The target system trains its activity recognition system based on T_T (e.g. running a feature extraction and selection process and training a classifier based on T_T). At this point the target system is ready to operate on the data of domain T.

3.3 Transfer of Activity Models

In this case, the recognition system devised for the source domain relies on activity models M_S, i.e. the parameters of the recognition system, including the selected set of features, the trained classifiers, etc. The goal is for the target system to use the same activity models ($M_T = M_S$). Therefore, the transfer process basically consists in copying M_S to the target model. However, to be capable of using M_T the target system requires its signals to be translated into the source domain. To do so, the target system uses $\Psi_{T\rightarrow S}$ to translate the sensor signals of domain T ($\mathbf{x_T}(t)$) into domain S ($\hat{\mathbf{x}}_\mathbf{S}(t)$) prior to applying the activity recognition model. Here again, $\Psi_{T\rightarrow S}$ is obtained through system identification. The complete activity model transfer architecture is shown in Fig. 2.

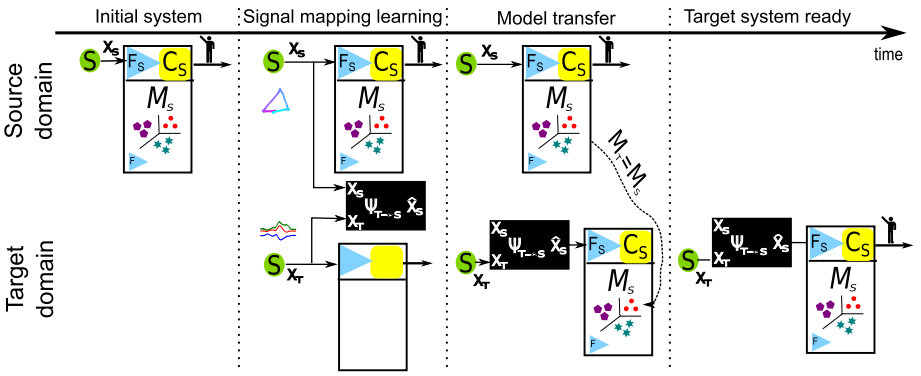


Fig. 2 Architecture for the transfer of activity models. From left to right, initially a fully operational activity recognition system is identified, hereafter the source system. Such system is defined through activity models M_S, which basically represent the feature set F_S and trained classifier C_S. Then, a mapping function $\Psi_{T\rightarrow S}$ between target and source domains is discovered through system identification. Next, the source activity models M_S are transferred to the target domain to define the target activity models M_T so that both models are the same. These activity models also define the target activity recognition system. Finally, the target system continuously translates its signals $\mathbf{x_T}(t)$ into the source domain as to estimate the equivalent target signals $\hat{\mathbf{x}}_\mathbf{S}(t)$ to operate the new recognition system. Note: the depicted signals may for example represent position (source domain) and acceleration (target domain)

4 Evaluation

4.1 Experimental Setup

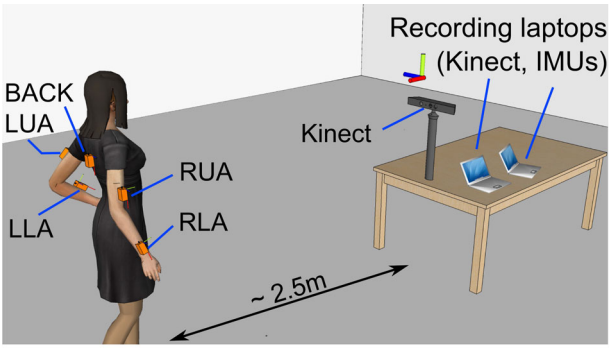
To evaluate the capabilities of the proposed transfer methods a multimodal setting is considered. The test bench namely consists of a human–computer interaction (HCI) gesture scenario in which activity recognition capabilities are transferred between sensors of the same modality (body-worn and body-worn sensors, i.e. identical transfer) and different modality (body-worn and ambient sensors, i.e. cross transfer). The body-worn sensors are five inertial measurement units (IMU) from which only the acceleration data is considered since this sensing modality is predominantly used in wearable activity recognition. The ambient sensor is a consumer vision-based skeleton tracking system (Microsoft Kinect) that provides the position of the body joints. The fact that the Kinect sensor normally builds on activity templates, and activity recognition systems based on IMU sensors operate on either activity templates or activity models, makes this a fairly appropriate setup to evaluate both transfer methods. The main technical specifications of the sensor setup are summarized in Table 1.

Kinect contains an 8-bit 640×480 RGB camera, an infrared (IR) LED projecting structured light (point cloud), and an IR camera. A dedicated integrated circuit processes the reflected structured IR light and computes an 11-bit 640 × 480 depth map in a range of 0.7–6 m. The cameras have a field of view of 57° horizontally and 43° vertically. The drivers fit a 15-joint skeleton on the depth map in real time and deliver 3D joint coordinates in millimeters measured from the Kinect center. Skeleton tracking is specified at ranges of 1.2–3.5 m, i.e. a 6 m^2 workspace, although longer ranges are achieved in practice. Kinect is interfaced over USB to a PC. The RGB and depth map videos and the joint coordinates are recorded at 30 Hz.

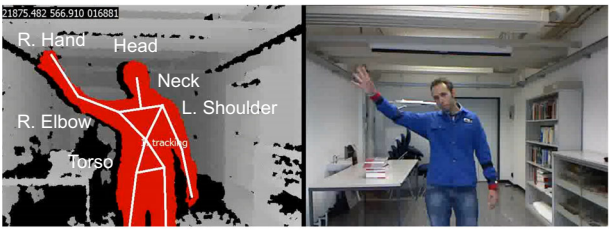
Five XSens IMUs [43] connected to a PC sense the upper body orientation. These IMUs contain gyroscope, magnetometer, and acceleration sensors combined with a Kalman filter to yield the device orientation in a world coordinate system in real-time. The raw sensor

Table 1 Main technical specifications of the sensor setup

Sensor	Measured data	Location	Sampling rate	Other specs
Kinect	3D position	15 body joints	30 Hz	1.2–3.5 m detection range
IMU	3D acceleration	Right/left lower/upper arm and back	30 Hz	38 × 53 × 21 mm size; 30 g weight



(a) Experimental Setup



(b) Kinect depth map and skeleton (left) and RGB (right)

Fig. 3 Kinect and IMU experimental setup. **a** Five IMUs and a Kinect capture the user's body movements. The IMUs are placed on the user's back (BACK), right upper arm (RUA), right lower arm (RLA), left upper arm (LUA) and left lower arm (LLA) respectively. **b** The Kinect sensor delivers a depth map, a color image, and a 15-joint skeleton of the user

data is collected at 30 Hz. As mentioned above, this evaluation only uses the 3D acceleration measured by the IMUs.

Kinect and IMU data are independently recorded. The acceleration data of the IMU is resampled to the regular Kinect sample comb to obtain an accurately synchronized dataset[1] comprising acceleration, position data, and labels. To that end, an infinite impulse response (IIR) elliptic filter is used, where the most linear phase response lies within the pass-band. Due to the nature of the gestures, the filter design parameters are set to 2 Hz corner frequency, 4 Hz stop frequency and 60 dB stop-band attenuation. A single subject performs five types of geometric gestures (circle, infinity, slider, triangle, square) with the right hand a total of 48 times in alternation. These gestures were selected because similar ones were demonstrated to be recognizable by wearable sensors [12] or Kinect [44]. They involve the lower and upper arm, which permits to assess the approach for the transfer of activity recognition systems between limbs. They are also diverse enough, which allows us to study if there

[1] Dataset available at http://orestibanos.com/datasets.htm.

exist preferential movements leading to a faster identification of the mapping between the two systems. The average ± standard deviation of the duration of each gesture classes are 2.97(89) ± 0.34(10), 3.28(98) ± 0.38(11), 2.23(66) ± 0.35(10), 3.12(93) ± 0.35(10) and 2.66(79) ± 0.48(14) seconds(samples). A 5 min long "idle" dataset, where the user performs infrequent low-amplitude arm movements and moves around, without any specific task is also recorded. The subject stands within 2–3 m from the Kinect sensor facing it within ±30° to avoid occlusions (Fig. 3a). Data labelling was performed on-the-fly and corrected later using video footage of the Kinect (Fig. 3b). Hand-claps at the start and end of the recording are also used for offline synchronization.

No constraints were placed on the way the gestures are executed beyond the subject trying the best to execute them similarly. However, it was observed that gestures were executed somewhat faster later in the recording and that the user position shifted away from the center of the camera field of view, until the user consciously moved back to the center. The left arm does not experience significant movement, thus the information monitored is not considered for this particular experiment. Moreover, since the ultimate goal would be to perform the transfer in a short time, for example, just by performing a reduced subset of informative gestures (ideally just one single gesture) to learn the mapping, this scenario is considered the most adequate for that purpose. One subject is just involved since the mapping should be learned irrespective of the person carrying out the movements.

4.2 Metrics and Methods

The capability of the proposed transfer methods is assessed through two metrics. First, the system identification performance is evaluated by assessing the quality or fit of the signal $\hat{\mathbf{x}}_\mathbf{T}$ (obtained by mapping $\mathbf{x}_\mathbf{S}$ to the target domain with the MIMO model) compared to the measured signal $\mathbf{x}_\mathbf{T}$. Second, we assess the accuracy with which the system can classify the gestures after transfer to the target domain T, compared to the accuracy in the source domain S, which is used here as a baseline.

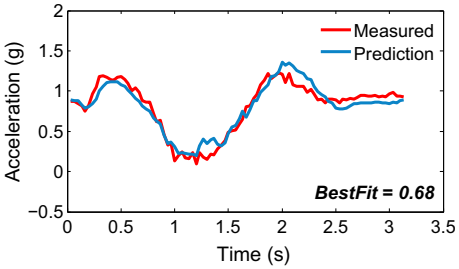
Mean square error (MSE) and root mean square error (RMSE) are commonly used to measure the fit in a regression problem. These metrics, however, are highly affected by the scale and offset of the signals. This may be partially addressed by their normalized variants (NMSE and NRMSE) and mean subtraction, which is essentially used in the *BestFit* metric we use here. The fit between the measured on-body acceleration $\mathbf{x}_\mathbf{T} = \mathbf{x}_\mathbf{I}$ and the predicted acceleration $\hat{\mathbf{x}}_\mathbf{T} = \hat{\mathbf{x}}_\mathbf{I}$, obtained by mapping the source signals to the target domain is calculated for each channel i:

$$BestFit_i = 1 - \frac{\left(\sum_{t=1}^{N}\left(x_T^{(i)}(t) - \hat{x}_T^{(i)}(t)\right)^2\right)^{\frac{1}{2}}}{\left(\sum_{t=1}^{N}\left(x_T^{(i)}(t) - \bar{x}_T^{(i)}\right)^2\right)^{\frac{1}{2}}} \tag{4}$$

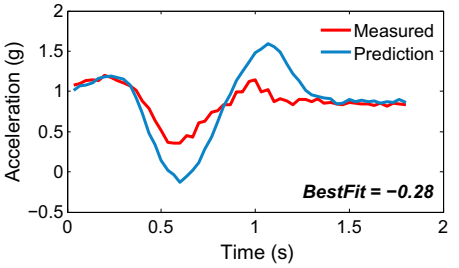
with N the number of signal samples, and $\bar{x}_T^{(i)}$ the mean over time of $x_T^{(i)}(t)$. The $BestFit_i$ is then averaged on all channels, resulting in a unified *BestFit* value. A *BestFit* of 1 indicates a perfect fit. Values above zero qualitatively indicate a good fit (Fig. 4). As *BestFit* tends to $-\infty$, the prediction differs more and more from the actual target signal.

Three kinds of MIMO mappings are evaluated in this work (Fig. 5):

– *Problem-domain mapping (PDM)*. This is a generic mapping learned on instances of all classes in equal proportions.



(a) Good fit



(b) Poor fit

Fig. 4 Comparison between the actual acceleration measured at the lower arm and the predicted after mapping from the hand position data sensed by Kinect: **a** for a circle and **b** a slider gesture. A good match between predicted and measured signals is obtained for *BestFit* values above 0

– *Gesture-specific mapping (GSM)*. This is a mapping learned on instances of a single class. It is used to analyze whether specific movements are more suited to identify the system dynamics.
– *Unrelated-domain mapping (UDM)*. This is a mapping learned from a sequence of samples from the "idle" dataset. It is used to assess mapping generalization across scenarios.

The models are trained with a minimum of data corresponding to roughly the duration of the longest gesture, which is set to 100 samples. Thus GSM and UDM are learned on 100 samples and PDM on 500 (as it requires data of each of the 5 gestures). The MIMO mappings are learned on a subset of the dataset and evaluated on the rest. The learning subset is obtained from a particular instance (GSM), by aggregating multiple activity instances (PDM), or obtained from the "idle" dataset (UDM).

To evaluate the accuracy of the transfer three non-overlapping parts (D_M, D_{C_t}, D_{C_v}) of the entire dataset ($D \subseteq D_M \cup D_{C_t} \cup D_{C_v}$) are used to: (1) learn the MIMO mapping (D_M); (2) train the source recognition system (D_{C_t}); and (3) test the transferred target recognition system (D_{C_v}). Source and target baseline evaluations are assessed by training and testing on data from the same domain, i.e. either $\mathbf{x_S}$ or $\mathbf{x_T}$ within $D_C \subseteq D_{C_t} \cup D_{C_v}$. Transfer evaluations use $\hat{\mathbf{x}}_\mathbf{T} = \Psi_{S \to T}(\mathbf{x_S})$ in D_{C_t} and $\mathbf{x_T}$ in D_{C_v} for the activity model transfer, and $\mathbf{x_S}$ in D_{C_t} and $\hat{\mathbf{x}}_\mathbf{S} = \Psi_{T \to S}(\mathbf{x_T})$ in D_{C_v} for the activity template transfer. The classifier training set D_{C_t} and test set D_{C_v} are defined by an instance-based random-seed 5-fold inner cross-validation process, repeated 100 times to ensure statistical robustness. D_M comprises a set of samples $\mathbf{x_S}$ and $\mathbf{x_T}$ obtained by aggregating multiple activity instances or obtained from the "idle" dataset. This process is random and repeated 20 times in an outer cross-validation process.

Two feature sets are used. Each instance is subdivided into 4 subwindows that capture the temporal dynamics and features are computed on them. Typical features used in activity recognition systems are employed. Concretely, FS1 corresponds to the mean of each axis (12 features) while FS2 is the maximum and minimum of each axis (24 features). The accuracy for segmented gestures recognition with a k-nearest neighbor (KNN) classifier is reported.

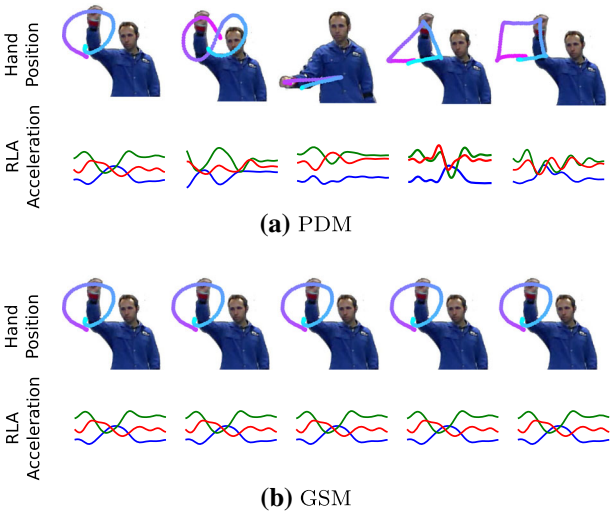


Fig. 5 Examples of MIMO mappings considered in this work: **a** problem-domain mapping (PDM) and **b** gesture-specific mapping (GSM). Unrelated-domain mapping (UDM) simply corresponds to a random sequence of arbitrary movements

KNN models have been proven to perform well in gesture recognition for both Kinect [45] and IMU [46,47]. A model similar to the one proposed in [48] is used here. The k-value for the KNN model is set to three given the good results shown in prior related works [20–22].

4.3 Transfer Between IMU and IMU

This section investigates the transfer of recognition capabilities from an existing activity recognition system operating on an IMU (I_S) to a new untrained system devised to operate on a different IMU (I_T). The translation between source and target relies on the identification of $\Psi_{I_S \to I_T}$ or $\Psi_{I_T \to I_S}$, which correspond to a 3-input (3D acceleration) 3-output (3D acceleration) MIMO mapping, thus within the same sensor modality (identical transfer). To fully capture the dynamics of the transformation, the mapping model is set to have ten tap delays. Thus, given $n_S = 3$, $n_T = 3$, and $q = 10$, a total of 108 parameters must be learned, which stems from the $(q + 1) \times n_T \times n_S$ coefficients of the polynomials and $n_T \times n_S$ static delays, as described in Sect. 3.1. Although the overfitting of the models potentially increase with the number of parameters, this is avoided here through the cross-validation procedure. The *System Identification* process characterizes as follows:

- A MIMO mapping $\Psi_{I_S \to I_T}$ from the 3D acceleration of the source IMU to the 3D acceleration of the target IMU is learned for the transfer of activity templates.
- The reverse MIMO mapping $\Psi_{I_T \to I_S}$ is needed for the transfer of activity models.

In Sect. 3 two methods that allow us to transfer activity recognition systems through the exchange of either activity templates (T) or activity models (M) were presented. Now, the specialization of these techniques to the IMU and IMU test case is described. Activity recognition systems based on IMU sensors may operate on both activity templates T and activity models M. Therefore, the two types of transfers are evaluated.

For the *Transfer of Activity Templates:*

- The source domain recognition system works on the 3D acceleration measured by the source IMU. It also stores the activity templates $T_S = T_{I_S}$ that are the 3D acceleration patterns for each gesture.
- $\mathbf{x_S} = \mathbf{x_{I_S}}$ is the 3D acceleration measured on the body by the source IMU.
- $\mathbf{x_T} = \mathbf{x_{I_T}}$ is the 3D acceleration measured on the body by the target IMU.
- $\hat{\mathbf{x}}_\mathbf{T} = \hat{\mathbf{x}}_{\mathbf{I_T}} = \Psi_{I_S \rightarrow I_T}(\mathbf{x_{I_S}})$ is the acceleration predicted on the body from the known source acceleration.
- After template translation, $T_T = T_{I_T}$ are the predicted 3D on-body acceleration patterns for each gesture and the corresponding class labels.
- The target recognition system is automatically trained at runtime on the templates T_T and eventually operates on the acceleration sensed by the target IMUs.

For the *Transfer of Activity Models:*

- The source domain recognition system works on the 3D acceleration sensed by the source IMU. It uses models $M_S = M_{I_S}$ for the recognition.
- $\mathbf{x_S} = \mathbf{x_{I_S}}$ is the 3D acceleration measured on the body by the source IMU.
- $\mathbf{x_T} = \mathbf{x_{I_T}}$ is the 3D acceleration measured on the body by the target IMU.
- $\hat{\mathbf{x}}_\mathbf{S} = \hat{\mathbf{x}}_{\mathbf{I_S}} = \Psi_{I_T \rightarrow I_S}(\mathbf{x_{I_T}})$ is the acceleration predicted on the body from the unknown source acceleration.
- After translation, the 3D acceleration signals of the target IMU are mapped to resemble the acceleration measured on the source IMU. In this way, the recognition models devised for the source IMU are used as-is by the target system ($M_T = M_S$), which now operates on the target IMU data.

The transfer among all possible pair combinations of the IMU sensors placed on the user's right lower arm (RLA), right upper arm (RUA), and back (BACK) are considered for evaluation. This leads to six cases of transfer of activity recognition from a source to target IMU, namely from RLA to RUA, RLA to BACK, RUA to RLA, RUA to BACK, BACK to RLA and BACK to RUA. For all these combinations the three MIMO mappings (PDM, GSM, and UDM) and both transfers (activity templates and activity models) are analyzed. These scenarios are devised to help investigate the potential of the transfer methods for sensor systems of the same modality (acceleration) but diverse domain (placed on close-by or unrelated body parts).

4.3.1 System Identification Performance

The *BestFit* computed from the evaluation of all possible pair combinations of mappings between RLA, RUA and BACK is depicted in Fig. 6. For example, the *BestFit* for the RLA to RUA mapping is computed between the acceleration measured at the lower arm and the acceleration predicted from the upper arm. From the results, the best fit tends to be obtained with PDM (Fig. 6a) followed by GSM (Fig. 6b) and UDM (Fig. 6c), in that order. This can be observed from the median (higher for PDM than GSM and UDM) and dispersion (lower for PDM than GSM and UDM) of the *BestFit* values.

PDM yields *BestFit* values which are above 0 in most cases, which was shown in Sect. 4.2 to represent a good fit. Moreover, for some mappings the *BestFit* distributions are close to 1, which corresponds to an almost perfect mapping. This could be expected, as the mappings are learned on the dynamics of all gestures. The results are also very promising for the GSM model. Once again, *BestFit* values greater than 0 are generally obtained but for the triangle

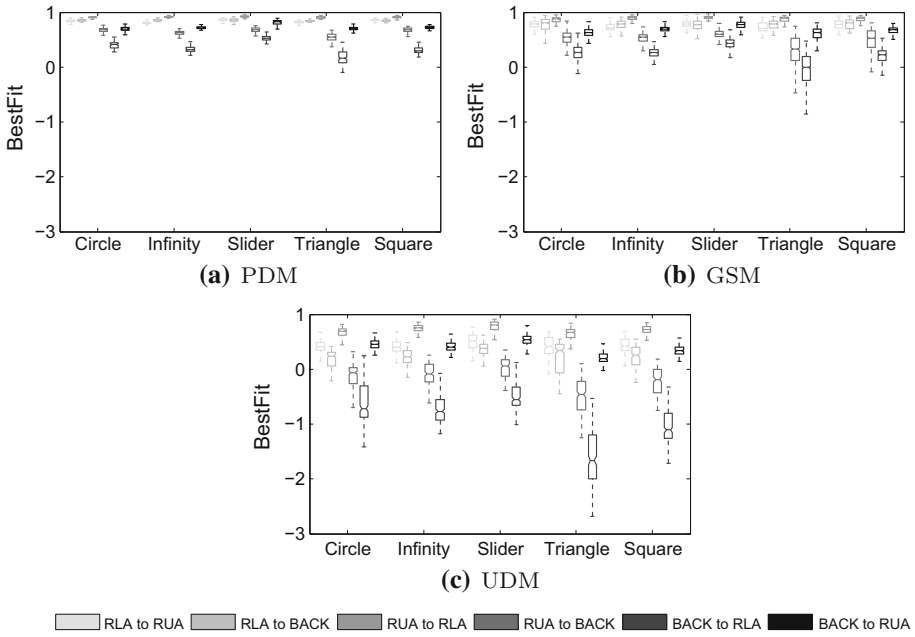


Fig. 6 Box plot of the *BestFit* distributions for all possible pair combinations of mappings between RLA, RUA, and BACK. The box plot represents the statistical distribution of the sample set (the central mark is the median, the edges of the box are the 25th and 75th percentiles, and the whiskers the most extreme results not considered outliers). *BestFit* for RLA to RUA, RLA to BACK, RUA to RLA, RUA to BACK, BACK to RLA, and BACK to RUA mappings are respectively represented by each box within each gesture group. **a** The mapping is trained on all gestures and the fit computed on the indicated gestures. **b** The mapping is trained on the indicated gesture and the fit computed on all of them. **c** The mapping is trained on data from another domain, and the fit is computed on the indicated gestures

and square gestures for some sensor combinations. A good fit is obtained for the rest of the gestures, thus either of them could in principle be enough to learn a mapping model. Finally, the UDM model provides the worst fit performances (Fig. 6c). Although high *BestFit* values are obtained for some mappings, these are in many cases below 0. The large dispersion in the results demonstrate that some of the movements performed while idling may be used for learning a mapping; however, many others fail to provide valuable information for capturing the dynamics of the physical system. This is consistent with the characteristics of the "idle" dataset since it has only rare occurrences of large amplitude limb movements. Nonetheless, this does not preclude that, a dataset from a domain not comprising the activities to recognize, but containing richer limb movements, might not be used to learn an adequate mapping model.

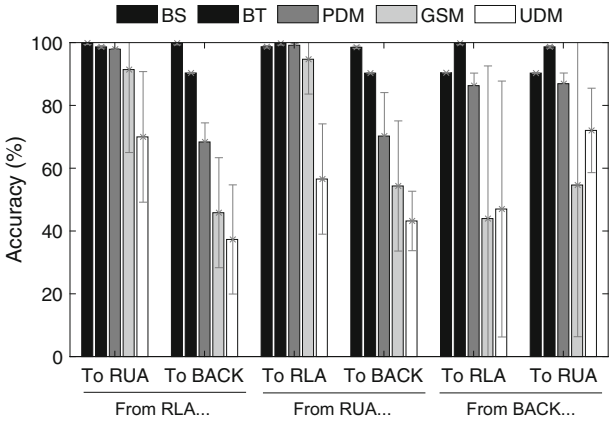
The most accurate fits are obtained between close-by limbs. Namely, mappings from the lower arm to the upper arm and vice versa obtain the highest *BestFit* values for all types of mappings. Likewise, mappings between upper arm and back prove to be good-enough. The fit worsens as the mapping model is computed between less related body regions (back to lower arm). This seems to be quite reasonable since lower arm and upper arm movements, and upper arm and back movements are more related to each other than lower arm and back ones.

The comparisons made among all models are exclusively based on the interpretation of the depicted box plots. A statistical analysis would be necessary to deem the encountered differences as statistically significant or not significant.

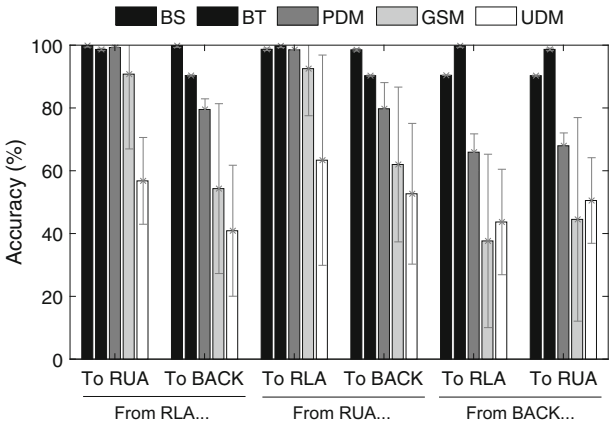
4.3.2 Transfer Accuracy

Classification accuracy baselines in the source (BS) and target domain (BT), and those after transfer to the target sensor are presented in Fig. 7 for all combinations of sensors and mappings. The reported results refer to the models using FS2, as this feature set is observed to be more sensitive to inaccurate signal mapping. The GSM mapping is learned on the "circle" gesture, which was identified as one of the best gestures to learn a mapping model.

Baselines represent the accuracy obtained by a recognition system that is trained and tested on the same sensor (no transfer). The baselines indicate that the gestures can be classified with an accuracy of 98% or more when using the lower-arm acceleration or the upper-arm acceleration. The high accuracy obtained for the back acceleration (baseline of about 89%)



(a) Transfer of activity templates



(b) Transfer of activity models

Fig. 7 Classification accuracy (average—bar—and confidence interval—whiskers—) for the transfer between two IMU systems with FS2. Transfer `from` a source system operating on **a** activity templates or **b** activity models `to` an untrained new system. Source and target systems are respectively identified through the X-axis. BS and BT indicate the baseline accuracies obtained with a system trained and tested only on the source and target data respectively. GSM mappings are based on the "circle" gesture, PDM mappings on all gestures, and UDM mappings on the idling scenario

indicates that torso movements are somewhat correlated with the execution of the gestures. This is a particular characteristic of this scenario, that likely does not generalize to other scenarios. The results after transfer must be assessed according to the performance drop from the baselines. The drop from BS indicates how much worse the system becomes after transfer. The drop from BT shows how much better a system devised specifically for the target domain would be.

The transfer between close-by limbs proves to be the most efficient. Best results are indeed for the transfer between upper arm and lower arm, with almost no drop observed after transfer for PDM and GSM. The performance obtained when using the PDM mapping model is similar to baseline, and reduces less than 7% at worst for the GSM mapping model. This implies that executing a single "circle" is sufficient to identify a reliable mapping model. Moreover, the transfer from the lower arm to the upper arm is practically similar as to when realized the other way around, thus direction independent. Transfers between distant body parts provide in general the worst results. The transfer between the lower arm and the back acceleration shows a variable drop from baseline depending on the mapping, namely 13–55% for the activity templates and 20–62% for the transfer of activity models. This is observed for both directions of the transfer. Better results are obtained when translating recognition capabilities between the upper arm and back systems. This demonstrates that the MIMO model learns more precisely the relations between linked domains. UDM appears to be not suitable for good transfers.

As for the quality of the transfer method, it is hard to draw any conclusion on which one performs better. The results are quite comparable between both transfer methods. Activity models seem to generally provide slightly better results, although this is not observed for all combinations.

4.4 Transfer Between Kinect and IMU

The transfer between sensors of different modality (cross transfer) is evaluated here. The transfer between Kinect (K) and IMU (I) relies on the identification of $\Psi_{K\to I}$, which is next characterized. $\Psi_{K\to I}$ is a 3-input (3D position) 3-output (3D acceleration) MIMO mapping with ten tap delays (q = 10, 108 parameters to learn). As for the transfer between IMUs, the tap delay is set to this value to ensure that the MIMO model captures the dynamics of the underlying relation that links both domains.

The specialization of the transfer of activity templates (T) and activity models (M) to the Kinect and IMU test case is now described. The Kinect recognition system is based on T, while the IMUs are seen to employ M. Accordingly, transfer learning from Kinect to IMUs will make use of the transfer of activity templates whilst the transfer from IMUs to Kinect will be performed through the transfer of activity models.

First of all, the signal mapping from Kinect (position) to IMU (acceleration) is identified. The *System Identification* process characterizes through:

- A MIMO mapping $\Psi_{K\to I}$ from the 3D Kinect joint position to the 3D acceleration is learned. As acceleration is the second derivative of position, this requires the MIMO mapping to realize at least a 2nd order differentiation.
- The learned model can be used both to translate from Kinect to IMU, and from IMU to Kinect, thanks to the two transfer models. The reverse MIMO mapping is not required.

For the *Transfer of Activity Templates (from Kinect to IMU):*

- The source domain recognition system works on the 3D position coordinates. It also stores the activity templates $T_S = T_K$ that are the 3D joint coordinates for each gesture.

- $\mathbf{x_S} = \mathbf{x_K}$ is the 3D joint position measured by the Kinect sensor (source).
- $\mathbf{x_T} = \mathbf{x_I}$ is the 3D acceleration measured on the body (target).
- $\hat{\mathbf{x}}_\mathbf{T} = \hat{\mathbf{x}}_\mathbf{I} = \Psi_{K \to I}(\mathbf{x_K})$ is the acceleration predicted on the body from the known joint position.
- After template translation, $T_T = T_I$ are the predicted 3D on-body acceleration patterns for each gesture and the corresponding class labels.
- The target recognition system is automatically trained at runtime on the templates T_T and finally operates on the acceleration sensed by the IMUs.

For the *Transfer of Activity Models (from IMU to Kinect):*

- The source domain recognition system works on the 3D acceleration sensed by an IMU. It uses models $M_S = M_I$ for the recognition.
- $\mathbf{x_S} = \mathbf{x_I}$ is the 3D acceleration measured on the body (source).
- $\mathbf{x_T} = \mathbf{x_K}$ is the 3D joint position measured by the Kinect sensor (target).
- $\hat{\mathbf{x}}_\mathbf{S} = \hat{\mathbf{x}}_\mathbf{I} = \Psi_{K \to I}(\mathbf{x_K})$ is the acceleration predicted on the body from the joint position.
- After translation, the 3D joint coordinates of the Kinect are mapped to show like acceleration. The recognition models devised for the IMU are used as-is by the target system ($M_T = M_S$) that now operates on the Kinect data.

According to the devised experimental setup, six cases of transfer of activity recognition from a source to target domain are studied. Three are cases of transfer of an existing ambient activity recognition system operating on the joint positions delivered by the Kinect hand towards a new system which will use body-worn accelerometers for activity recognition. The on-body sensors are worn either on the lower arm, the upper arm, or the back. This is done with the transfer of activity templates. The other three cases transfer an existing wearable activity recognition system operating on on-body accelerometers towards an ambient system which will use the joint position of the hand delivered by Kinect. The on-body sensors are worn either on the lower arm, the upper arm, or the back. This is done using the transfer of activity models. These scenarios are devised to help investigate the potential of the transfer methods for sensor systems of diverse modality (position/acceleration) and of diverse domain (centered on close-by or unrelated body parts).

4.4.1 System Identification Performance

The *BestFit* computed between the acceleration measured at the lower arm, upper arm, and back, and the acceleration predicted from the Kinect hand position is presented in Fig. 8. Most precise fits are obtained for the PDM approach (Fig. 8a) followed by GSM (Fig. 8b) and UDM (Fig. 8c), in that order. This can be easily detected by looking at the median (higher for PDM than GSM and UDM) and dispersion (lower for PDM than GSM and UDM) of the *BestFit* values.

The results obtained for the PDM case show that a single mapping model is suitable for a wide range of gestures when it is learned on one gesture of each class. This is consistent with the results obtained for the IMU to IMU case, as the mappings are learned on the dynamics of all gestures. The learning of a good mapping is also possible based on one gesture of a given class (Fig. 8b). In such a case, the best unique gesture to learn a mapping model seems to be the circle, followed by the triangle or square, yielding the highest *BestFit* values. This indicates that these gestures are sufficient for the model to capture the dynamics of the physical system. The fitness for the slider gesture is much worse. Despite the low dynamism of this movement, the mapping obtained for this gesture is not necessary poorer classification-wise. Considering the *BestFit* metric definition, the lower the absolute deviation the lower

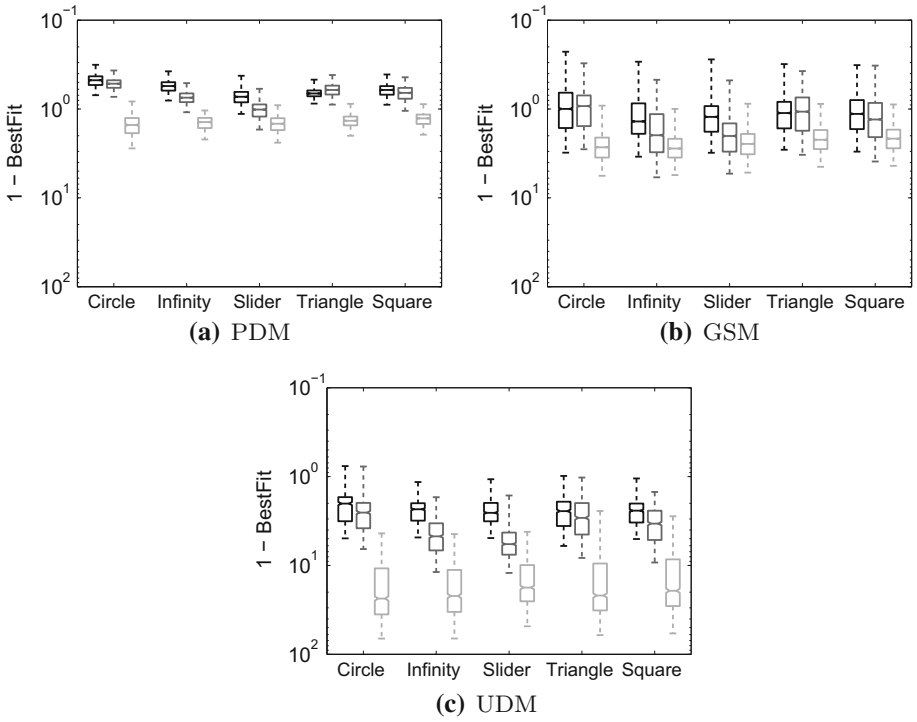


Fig. 8 Logarithmic box plot representing the statistical distribution of the sample set (the central mark is the median, the edges of the box are the 25th and 75th percentiles, and the whiskers the most extreme results not considered outliers) of $1 - BestFit$ between the acceleration measured at the lower arm, upper arm, and back (first, second and third box within each gesture group) and the acceleration predicted at that location from the position of the hand measured by Kinect. **a** The mapping is trained on all gestures and the fit computed on the indicated gestures. **b** The mapping is trained on the indicated gesture and the fit computed on all of them. **c** The mapping is trained on data from another domain, and the fit is computed on the indicated gestures

the dissimilarity tolerated between target and prediction. In other words, since the target has a low amplitude the prediction errors are enhanced, which is particularly evident for this gesture.

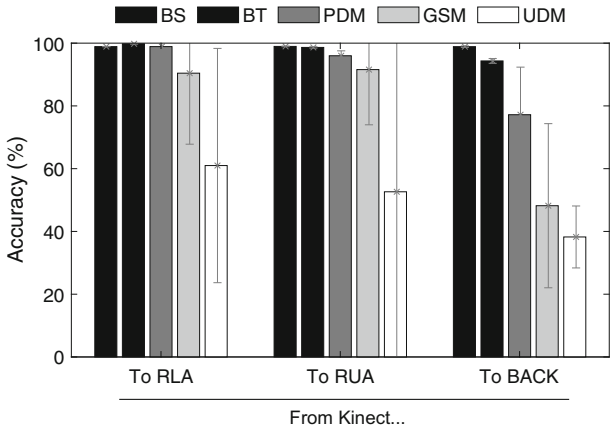
UDM does not achieve a good mapping (Fig. 8c). The movements in the "idle" dataset, with only rare occurrences of larger amplitude limb movements, are insufficient to model the dynamics of the physical system. Nevertheless, a dataset from a domain that does not comprise the activities to recognize but which contains richer limb movements may be used to learn a good mapping model. In either case, it will be seen later that this can be compensated by using more data (Fig. 10). The fit worsens as the mapping model is computed between less related body regions, i.e. hand to upper arm or hand to back. Nevertheless, the fit between hand position and upper arm acceleration is close to that of the hand position to lower arm acceleration. This means that the movement of the upper arm is relatively well predicted from the movement of the hand. This suggests that this approach may be applicable to transfer activity models across close-by and related limbs also for cross domains. The back acceleration is hardly predictable from the hand position, which is consistent with the explanation given for the IMU to IMU transfer in this respect.

The comparisons made among all models are exclusively based on the interpretation of the depicted box plots. A statistical analysis would be necessary to deem the encountered differences as statistically significant or not significant.

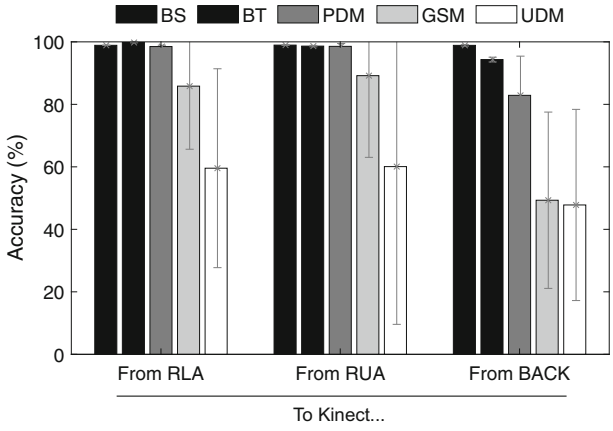
4.4.2 Transfer Accuracy

Baselines in the source (BS) and target domain (BT) and classification accuracies after transfer to the target domain are shown in Fig. 9. FS2 and GSM mappings based on the "circle" gesture are also used here as for the IMU to IMU case. Almost perfect accuracies are obtained for the baseline classification based on the hand position data registered through the Kinect sensor, fairly aligned with the already reported excellent results obtained for baselines based on the lower-arm acceleration and the upper-arm acceleration respectively.

In the transfer between hand position and lower or upper arm acceleration, the PDM and GSM models tend to perform just as well. The best results are obtained when transferring



(a) Transfer of activity templates



(b) Transfer of activity models

Fig. 9 Classification accuracy (average—bar—and confidence interval—whiskers—) for the transfer between ambient (Kinect) and IMU systems with FS2. Transfer `from` a source system operating on **a** activity templates or **b** activity models `to` an untrained new system. Source and target systems are respectively identified through the X-axis. BS and BT indicate the baseline accuracies obtained with a system trained and tested only on the source and target data respectively. GSM mappings are based on the "circle" gesture, PDM mappings on all gestures, and UDM mappings on the idling scenario

between hand position and lower-arm acceleration and between hand position and upper-arm acceleration for the PDM model. Performance drops for these transfers are below 4% with respect to BS. The direction of the transfer does not affect the results significantly. Given the results for the GSM approach, a single "circle" is also sufficient to identify a mapping model that leads to a transfer with performance drops below 8% from BS. On the contrary, the transfer between the hand position and the back acceleration shows a large drop with respect to BS ranging from 20 to 60% for all types of mappings. This is consistent with the low *BestFit* obtained when attempting to predict the back acceleration from hand position. Also, inline with the *BestFit*, the UDM models appear unsuitable for the transfer. As for the IMU to IMU case, the results are quite comparable between both transfer methods. Activity templates seem to provide subtly better results, although this is not observed for all combinations.

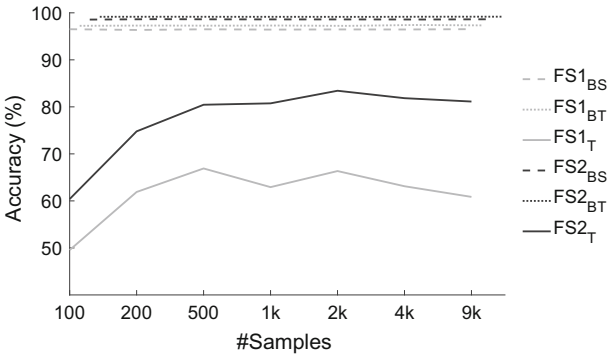
While GSM and PDM have shown to yield good mappings based on sets of 100 (3.33 s) and 500 (16.67 s) samples respectively, UDM models fail to do so for such a limited amount of learning data. An evaluation for different amounts of learning data is therefore performed to understand how much data might be required to develop a good transfer based on UDM. The results of such evaluation are presented in Fig. 10. The UDM mapping improves when learned on more "idle" data. With 2000 samples (67 s), the performance is about 15–30% below the corresponding baselines for FS2 and FS1 respectively. This suggests that, with sufficient data, a dataset from an unrelated domain allows the MIMO mappings to capture the dynamics of the physical system. This has practical benefits since "unrelated" domain data can easily be acquired in the background, whenever the user is in the sensing range of the source and target sensor systems. Even more, if the user's movements are of higher-amplitude than in the "idle" dataset, previous results show that the time to learn the mapping model can be reduced significantly (see for example *BestFit* for GSM). The difference between FS1 and FS2 highlights that an automatic selection of better features by the source or target system may lead to improved results. Hence, the reported results are a lower bound on the potential performance of the transfer.

5 Discussion

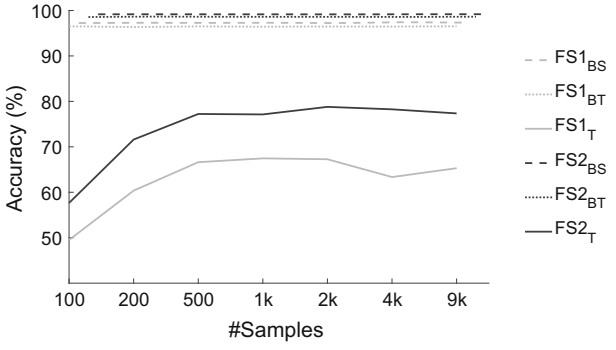
5.1 Transfer Method Benefits

IoT based activity recognition systems are subject to changes in the sensor setup due to upgrades and maintenance of the sensing ecosystem (i.e. replacement or addition). To make use of the newcomer sensors, standard activity recognition systems normally require complete retraining. Such training typically involves the collection of a new (big) dataset, which turns to be quite impractical in real-world applications. The transfer approach proposed in this work serves to translate the recognition capabilities of an existing system to a newly introduced untrained system, a transfer that may be performed at run-time and without requiring expert or user intervention.

The evaluation of the proposed approach has been demonstrated to succeed for transfers between same and different modalities. The model is capable of capturing the underlying relation between systems of identical modality translated or rotated with respect to each other. Similarly, the proposed method proves to be capable of discovering the physical relation between cross domains. Although this has been here demonstrated for Kinect-based and accelerometer-based systems, the approach itself is generic and can be applied to other sensing systems.



(a) Kinect to IMU transfer



(b) IMU to Kinect transfer

Fig. 10 Influence of the amount of "idle" data used to learn the UDM mapping on the classification accuracy for a transfer **a** from Kinect hand position to acceleration at the lower arm and **b** from acceleration at the lower arm to Kinect hand position. Evaluations are performed on both feature sets 1 and 2 ($FS1_T$, $FS2_T$). BS and BT represent the source and target baselines

The approach shows to scale well with the number of classes. A mapping learned with one instance of a single class (GSM) performed well on the prediction of the signals of other gestures. This suggests that the mapping approximates the physical relations between the sensing systems and not just a data-dependent transformation, thus independently of the gestures. Although offline experiments are performed in this work, the approach is also applicable for continuous recognition. Therefore, once a mapping model is identified the target signal can be transformed to look like the signal used by a source system with recognition capabilities.

The learning approach has the advantage of approximating the signal mapping even with low-variance data unrelated to the activities of interest (UDM). However, a longer coexistence time is required, which in turn means that more data is necessary to learn the mapping. This is rather practical given the fact that "unrelated" domain data can be easily acquired in the background, whenever the user is in the sensing range of source and target sensor systems. For example, when a new sensor is added or replaced, the learning may be performed while the user executes their daily routines, without interfering with their normal duties. A better mapping is normally learned however from the execution of movements highlighting the physical relation between the sensor systems.

The proposed transfer approach may prove useful for personalising activity recognition systems by translating generic activity models to the specific sensor modalities that a given

user has. Moreover, some fields such as video gaming could highly benefit from this type of learning since multi-sensing platforms not envisioned during design time may be easily learned and used to enhance user experience. For example, users may play with Kinect when staring in front of the camera but keep playing when they get out of the sensing range of this sensor. This can be achieved by transferring recognition capabilities to a smartwatch or smartphone (in line with the ubiquitous gaming experience offered by video game consoles such as Nintendo Switch). Transfer learning may also facilitate video game developers tasks when porting functionality and playability from a given sensing platform (e.g. Microsoft Kinect) to a different but related sensing equipment (e.g. Playstation Move).

5.2 System Identification

In a predefined, fixed, and non-changeable sensor ecosystem one could prefer a WB over a BB mapping. However, in the usually varying IoT ecosystems, a WB mapping would prove quite impractical as it would not generalize to modalities or configurations unforeseen by the system designer. The proposed approach allows us to take advantage of additional sensors as they become available and to learn the mapping without expert intervention. If the needed transformations become more complex than those presented in this work, the approach is still valid and eventually the BB can be chosen among a more powerful set of models, e.g. time-delay neural networks [49] or non-linear ARMA [38], addressing non-linearities or time-variance. These models support non-linear transformations, such as those needed when a sensor changes properties over time. Nevertheless, more complex transformations likely need longer coexistence time between source and target to estimate the model parameters. Besides, non-linear models are more prone to overfitting and learning of noise and signal artifacts, which typically appear on the registered data. The possibility of updating the system identification model instead of relearning it from scratch is also of much interest in our context. Online non-linear system identification models are meant to do that [39].

Gray-box (GB) models, an intermediate approach between BB and WB modeling could be also considered. GB models combine the best of WB and BB approaches [50], i.e. knowledge-based modeling through mathematical equations that describe the physical process and parametrized modeling where parameters are estimated directly from the measured data. The limitations of WB may still apply to GB, thus the extent to which these models may be applicable in our context depends on how the GB model is specifically defined.

The mapping between the modalities considered here might look trivial at first glance. For example, bodily accelerations are related by biomechanics constraints, and acceleration can be calculated from position. However, the acceleration measured by the IMUs is given in a local, time-varying frame of reference. The frames of reference of the sensors are not identical and their relative orientation may change over time. Moreover, these modalities exhibit complex transfer characteristics between each other. In fact, there is a dynamic relation between acceleration and position involving current and past samples. Thus, this setup allows us to conduct a non-trivial, yet manageable analysis of our method.

System identification models are shown to cope with the complexity of the signal mapping. Nevertheless, it must be present the importance of synchronization among source and target signals during the mapping learning process. Although the models have a certain tolerance (not analyzed here) a significant delay or jitter might prevent us from obtaining a functional transfer model. Therefore, synchronization among signals as well as resampling and downsampling techniques should be somehow considered at the point of need.

5.3 Transfer of Activity Templates vs. Transfer of Activity Models

The choice of transfer type will normally depend on the characteristics of the source recognition system (i.e. works on templates or uses a trained model). Moreover, the proposed transfer models involve different computational complexities in their operating phases and present different storage requirements, important aspects when dealing with IoT sensor ecosystems. In the template transfer approach, the activity templates are translated once, and then the target system incurs no additional computational load beyond feature computation and classification. This transfer requires, however, the source system to store activity templates, which in any case does not demand a large amount of space (few kilobytes in our case). Hence, this transfer is well suited for an ambient source system, which tends to have fewer restrictions on memory, and a wearable target system, which is more prone to suffer from storage constraints. On the contrary, the transfer of activity models requires that the target sensor signals are seamlessly translated from the target to the source domain. This increases the computational load on the target but the mapping complexity is low and easily benefits from streaming single instruction multiple data computation. The storage requirement is lower since only the activity models need to be stored. This is well suited for a wearable source and an ambient target system. The proposed transfer approaches also differ in whether the source signal is mapped to the target, or vice versa. If a mapping model exists both ways, as it happens to occur for the IMU to IMU case, then the choice of the transfer is based on computational and memory requirements. If the mapping model is more accurate in one way, then the transfer that uses this mapping should be favored.

5.4 Open Issues and Future Opportunities

The gestures considered in this work are performed relatively slowly, thus leading to low intensity accelerations (below 1.3 g). Hence, the accelerometer readings are mainly due to the time-varying angle formed between the sensor and the gravity vector. If higher dynamics are present, the mapping has to learn how to calculate the transformed static part (due to gravity) and the dynamic part (due to movements) and the transformations are not necessarily identical. For example, accelerometers mounted on two different positions of the forearm might have the same orientation with respect to gravity but the dynamic acceleration will in principle grow with the distance from the elbow. Furthermore, accelerometers measure the data within a local frame of reference while an external positioning system like Kinect collects data referred to a fixed world frame. Thus, the rotation matrix needed for transformation is dependent on the body posture itself in a non-linear manner. In other words, the signal mapping would have to include not only a second derivative but also a rotation which depends non-linearly on the body posture. The linear MIMO model can only approximate the second derivative and a fixed rotation, which would be an average rotation. This may become an issue with more ample movements but in our dataset the relative rotation of the frames of reference was limited for most gestures ($\pm$ 30–40°). Only for the slider gesture, the lower arm rotates by almost 90° at the extreme of the movement, compared to the starting position.

This very limitation is further observed in the event of a sensor rotation (e.g. the accelerometer is rotated with respect to its original position). The linear MIMO model can only be learned in streaming for a specific sensor orientation; thus, the model must be generally retrained in case of sensor rotations. The use of features invariant to rotations for classification may help in some cases to deal with these effects.

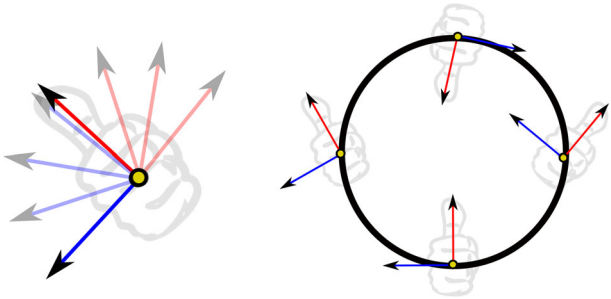


Fig. 11 Example of torsion along the forearm (left). Example of torsion when performing a gesture (right). The superimposed axes show the change of the local frame of reference of an IMU sensor placed on the wrist. Torsion may not be sensed through the 3D-position Kinect model

Some data types are particularly prone to be contaminated or affected by anomalies of different kind. For example, Kinect, or any other video-based tracking system, may be affected by occlusions. Likewise, magnetic field signals can be also distorted when IMUs are in the presence or proximity of ferromagnetic surfaces. Such situations may potentially lead to inconsistent or noisy data. As it has been shown, little data is generally needed to learn the mapping, which may be available in-between occlusions or when sufficiently far from the disturbance. Moreover, in presence of anomalies, the *BestFit* decreases, indicating that a proper mapping cannot be performed. This information can be used to determine when the data is of a quality such that allows for good learning. The transfer can be attempted as soon as the *BestFit* improves. The use of signal preprocessing techniques may be particularly recommended for those cases where the disturbance remains over time.

There can exist some subtle movements that may not be sensed by certain modalities. For example, Kinect cannot detect torsions of hand and forearm (e.g. in gestures like turning a knob or tightening a screw). Such torsions result into changes of the local IMU frame of reference with respect to the camera world coordinates by Kinect. Fig. 11 (left) shows an illustrative example where the arm is rotated along its generation axis but no movement is appreciated from the body joint camera model. This may be part of a specific gesture or movement as shown in Fig. 11 (right). In both cases, diverse acceleration signals could be expected. In contrast, torsions are easily sensed by gyroscopes and accelerometers, meaning that the expected transfer performance is modality- and gesture-dependent. This makes this approach well suited to opportunistically improve the mapping models by taking advantage of additional sensor modalities as they become available.

The problem of having diverse possible representations for a gesture or movement in one domain that may correspond to a unique representation in the other domain also applies the other way around. Let us consider a subject that performs a gesture in a given position, and then moves to another distant position (yet in the range of Kinect) and executes an exact replica of the gesture. The registered inertial signals are seen to be identical for both executions but different for Kinect. A small set of tests, not reported in this work, were performed to analyze how the mapping model deals with this situation. First, all the Kinect 3D coordinates of the hand signals were referred to a similar origin randomly selected. No significant difference was encountered in terms of both fitness and transfer accuracy. Moreover, it was evaluated the removal of the body model center of mass to provide a common reference. Again, no better results were obtained than for the original data. From a mathematical point of view this is expected since the derivatives (position to acceleration) filter out the offset between both

spacial gesture realizations. This confirms once more the capacity of the mapping model for capturing the physics of the underlying systems relation.

While the proposed approach was originally conceived to transfer activity recognition capabilities between wearable and ambient sensor-based systems, it could be also interesting to explore its use in between device-free activity recognition systems (e.g. between Wi-Fi-based systems). These systems are becoming increasingly popular as they do not require the user to wear a specific sensor and they also deal gently with some well-known limitations of video-based systems (e.g. line of sight conditions). Non-linear transfer learning approaches are possibly most useful for these technologies given the intricate nature of some of the signals considered in this type of contactless activity recognition.

The proposed approach has been only tested for one subject and a specific HCI scenario. While the approach is defined to be user- and setting-agnostic, it would be interesting to corroborate our findings by testing it with more subjects, other types of activities, and different application scenarios. A recent work [51] could be considered to that end as it provides a practical means to generate cross-domain data for different subjects and scenarios by simulating IMU data from monocular RGB videos (e.g. existing YouTube videos). Another interesting aspect that could be explored in future work is the impact of transferring ill-defined or poor-performing models. The use of advanced semantic representations of the activity models can in principle help define the type and performance of transfer learning approaches, as well as its robustness against negative transfer effects [52].

6 Conclusions

Present and future IoT ecosystems represent unparalleled opportunities for human activity recognition, where highly and densely sensorized setups can collect a myriad of multimodal data from users and their context. However, such sensor setups are prone to a variety of changes. Obsolete or damaged sensors may be replaced with sensors of different characteristics or new sensors added as part of equipment maintenance and upgrades. Moreover, sensors availability may also vary during the normal course of a person's day depending on their particular context (e.g. a user wears a smartband at the gym, uses a webcam at work, or interacts with proximity sensors at home). These sensors may not be capable of specific activity recognition since either they may not have associated activity models or are originally devised for other purposes. Specific training is then required for these newcomer systems to become usable for the recognition of the activities of interest. To build the recognition models, the collection of new experimental data is in principle required, which happens to be unpractical and unapproachable in realistic scenarios. Transfer learning is devised here as the perfect means to realize this training in a suitable fashion.

Two transfer modes are proposed for the translation of activity recognition systems that operate either on activity templates or activity models. The transfer learning models are evaluated in a multimodal gesture recognition setting consisting of a vision-based skeleton tracking system and body-worn accelerometers. For this evaluation, two scenarios of transfer learning have been analyzed.

Firstly, the transfer between sensors of the same modality (identical transfer) is studied, namely between IMUs (accelerometers) placed in different body parts. System identification techniques are shown to successfully learn a linear MIMO model that maps the 3D accelerations sensed by source and target IMUs. As few as a single gesture (3 s) of data is enough to learn a mapping model that captures the dynamics of the physical system. This allows us to

transfer quite well the recognition capabilities of systems that operate on close-by or related sensors. The IMU to IMU translation across adjacent limbs (here lower arm and upper arm) achieves recognition accuracies above 97%, which is less than 2% below the accuracy of the original system. The quality of the transfer proves to be independent of the direction of the transfer.

Secondly, the transfer between sensors of different modality (cross transfer) is pursued. A linear MIMO model that maps 3D position sensed by a depth video system (Kinect) to the 3D acceleration measured on-body by IMUs can be achieved through system identification. Here again, one gesture is in principle enough to learn a mapping model that captures the dynamics of the physical system. The approach generalizes to unseen movements when the user is active; however, more data is required to learn this mapping when the user is idle, yet much less than other transfer approaches.

The Kinect to IMU and IMU to Kinect translation achieves a recognition accuracy of 95%, and is less than 4% below the accuracy of the initial system. These results hold even when the translation is to an adjacent limb (e.g. Kinect hand to IMU on the upper-arm).

This work supports the translation of activity recognition capabilities between sensor modalities without user or system designer's intervention. This is an important characteristic for activity recognition in IoT open-ended environments, where characteristics and availability of sensors may change over time. The proposed transfer method is generic and could be applied to other type of sensors, for example between a gyroscope and an inclinometer embedded into smart clothing. Future work should evaluate the approach with more complex sensing environments and modalities, where non-linear mappings and transfer of more sophisticated knowledge representations might be necessary.

Acknowledgements This work has been partially supported by the Spanish Ministry of Science, Innovation and Universities (MICINN) Projects PGC2018-098813-B-C31 and RTI2018-101674-B-I00 together with the European Fund for Regional Development (FEDER).

References

1. Lin J, Wei Yu, Zhang N, Yang X, Zhang H, Zhao W (2017) A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J 4(5):1125–1142
2. Chen L, Nugent CD (2019) Human activity recognition and behaviour analysis. Springer, Berlin
3. Lukowicz P, Hanser F, Szubski C, Schobersberger W (2006) Detecting and interpreting muscle activity with wearable force sensors. In: Pervasive computing, pp 101–116
4. Amft O (2010) A wearable earpad sensor for chewing monitoring. In: IEEE sensors conference, pp 222–227
5. Banos O, Damas M, Pomares H, Prieto A, Rojas I (2012) Daily living activity recognition based on statistical feature quality group selection. Expert Syst Appl 39(9):8013–8021
6. Wang F, Liu J (2010) Networked wireless sensor data collection: issues, challenges, and approaches. IEEE Commun Surv Tutor 13(4):673–687
7. Roggen D, Troester G, Lukowicz P, Ferscha L, Millan JR, Chavarriaga R (2013) Opportunistic human activity and context recognition. Computer 46(2):36–45
8. Guo B, Zhang D, Wang Z, Zhiwen Yu, Zhou X (2013) Opportunistic IoT: exploring the harmonious interaction between human and the internet of things. J Netw Comput Appl 36(6):1531–1539
9. Villalonga C, Pomares H, Rojas I, Banos O (2017) MIMU-wear: ontology-based sensor selection for real-world wearable activity recognition. Neurocomputing 250:76–100
10. Pan SJ, Yang Q (2009) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
11. Lester J, Choudhury T, Borriello G (2006) A practical approach to recognizing physical activities. In: Proceedings of pervasive computing, pp 1–16
12. Chavarriaga R, Bayati H, Del Millán J (2013) Unsupervised adaptation for acceleration-based activity recognition: robustness to sensor displacement and rotation. Pers Ubiquit Comput 17(3):479–490

13. Foerster K, Roggen D, Troester G (2009) Unsupervised classifier self-calibration through repeated context occurences: is there robustness against sensor displacement to gain? In: International symposium on wearable computers, pp 77–84, Linz, Austria
14. Kunze K, Lukowicz P (2008) Dealing with sensor displacement in motion-based onbody activity recognition systems. In: International conference on ubiquitous computing, pp 20–29
15. Kunze K, Bahle G, Lukowicz P, Partridge K (2010) Can magnetic field sensors replace gyroscopes in wearable sensing applications? In: International symposium on wearable computers
16. Kunze K, Lukowicz P (2014) Sensor placement variations in wearable activity recognition. IEEE Pervasive Comput 13(4):32–41
17. Kunze K, Lukowicz P, Partridge K, Begole B (2009) Which way am i facing: inferring horizontal device orientation from an accelerometer signal. In: International symposium on wearable computers, pp 149–150
18. Yurtman A, Barshan B (2017) Activity recognition invariant to sensor orientation with wearable motion sensors. Sensors 17(8):1838
19. Yurtman A, Barshan B, Fidan B (2018) Activity recognition invariant to wearable sensor unit orientation using differential rotational transformations represented by quaternions. Sensors 18(8):2725
20. Banos O, Damas M, Guillen A, Herrera L-J, Pomares H, Rojas I, Villalonga C (2015) Multi-sensor fusion based on asymmetric decision weighting for robust activity recognition. Neural Process Lett 42(1):5–26
21. Banos O, Damas M, Pomares H, Rojas I (2012) On the use of sensor fusion to reduce the impact of rotational and additive noise in human activity recognition. Sensors 12(6):8039–8054
22. Banos O, Toth MA, Damas M, Pomares H, Rojas I (2014) Dealing with the effects of sensor displacement in wearable activity recognition. Sensors 14(6):9995–10023
23. Lam A, Roy-Chowdhury AK, Shelton CR (2010) Interactive event search through transfer learning. In: Asian conference on computer vision. Springer, Berlin, pp 157–170
24. Pan SJ, Tsang IW, Kwok JT, Yang Q (2011) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210
25. Nater F, Tommasi T, Grabner H, Van Gool L, Caputo B (2011) Transferring activities: updating human behavior analysis. In: IEEE international conference on computer vision workshops, pp 1737–1744
26. Al-Halah Z, Rybok L, Stiefelhagen R (2016) Transfer metric learning for action similarity using high-level semantics. Pattern Recogn Lett 72:82–90
27. Zheng VW, Hu DH, Yang Q (2009) Cross-domain activity recognition. In: International conference on ubiquitous computing, pp 61–70
28. Hu DH, Zheng VW, Yang Q (2011) Cross-domain activity recognition via transfer learning. Pervasive Mob Comput 7(3):344–358
29. Chiang Y, Hsu JY (2012) Knowledge transfer in activity recognition using sensor profile. In: International conference on ubiquitous intelligence computing and international conference on autonomic trusted computing, pp 180–187
30. Feuz K, Cook DJ (2014) Heterogeneous transfer learning for activity recognition using heuristic search techniques. Int J Pervasive Comput Commun 10(4):393–418
31. Ying JJ-C, Lin B-H, Tseng VS, Hsieh S-Y (2015) Transfer learning on high variety domains for activity recognition. In: ASE BigData and social informatics. ACM, pp 37:1–37:6
32. Feuz KD, Cook DJ (2015) Transfer learning across feature-rich heterogeneous feature spaces via feature-space remapping (FSR). ACM Trans Intell Syst Technol 6(1):3:1–3:27
33. Chen W-H, Cho P-C, Jiang Y-L (2017) Activity recognition using transfer learning. Sens Mater 29(7):897–904
34. Morales FJO, Roggen D (2016) Deep convolutional feature transfer across mobile activity recognition domains, sensor modalities and locations. In: Proceedings of the 2016 ACM international symposium on wearable computers. ACM, pp 92–99
35. Rokni SA, Ghasemzadeh H (2018) Autonomous training of activity recognition algorithms in mobile sensors: a transfer learning approach in context-invariant views. IEEE Trans Mob Comput 17(8):1764–1777
36. Calatroni A, Villalonga C, Roggen D, Troester G. (2009) Context cells: towards lifelong learning in activity recognition systems. In: European conference on smart sensing and context, pp 121–134
37. Calatroni A, Roggen D, Troester G (2011) Automatic transfer of activity recognition capabilities between body-worn motion sensors: training newcomers to recognize locomotion. In: International conference on networked sensing systems
38. Sjoeberg J, Zhang Q, Ljung L, Benveniste A, Delyon B, Glorennec P-Y, Hjalmarsson H, Juditsky A (1995) Nonlinear black-box modeling in system identification: a unified overview. Automatica 31(12):1691–1724
39. Hong X, Mitchell RJ, Chen S, Harris CJ, Li K, Irwin GW (2008) Model selection approaches for non-linear system identification: a review. Int J Syst Sci 39(10):925–946

40. Pota HR (1996) MIMO systems-transfer function to state-space. IEEE Trans Educ 39(1):97–99
41. Berndt DJ, Clifford J (1994) Using dynamic time warping to find patterns in time series. In: AAAI—KDD workshop, pp 359–370
42. Ye L, Keogh E (2011) Time series shapelets: a novel technique that allows accurate, interpretable and fast classification. Data Min Knowl Disc 22(1–2):149–182
43. Xsens Technologies B.V. (2009) XM-B technical documentation. http://www.xsens.com
44. Biswas KK, Basu SK (2011) Gesture recognition using microsoft kinect®. In: International conference on automation, robotics and applications. IEEE, pp 100–103
45. Hongyong T, Youling Y (2012) Finger tracking and gesture recognition with kinect. In: International conference on computer and information technology, pp 214–218
46. Foerster K, Biasiucci A, Chavarriaga R, Millan JDR, Roggen D, Troester G (2010) On the use of brain decoded signals for online user adaptive gesture recognition systems. In: International conference on pervasive computing, pp 427–444
47. Förster K, Monteleone S, Calatroni A, Roggen D, Tröster G (2010) Incremental KNN classifier exploiting correct—error teacher for activity recognition. In: International conference on machine learning and applications, pp 445–450
48. Cover TM, Hart PE (1967) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13(1):21–27
49. Yazdizadeh A, Khorasani K (2002) Adaptive time delay neural network structures for nonlinear system identification. Neurocomputing 47(1–4):207–240
50. Oussar Y, Dreyfus G (2001) How to be a gray box: dynamic semi-physical modeling. Neural Netw 14(9):1161–1172
51. Rey VF, Hevesi P, Kovalenko O, Lukowicz P (2019) Let there be IMU data: generating training data for wearable, motion sensor based activity recognition from monocular RGB videos. In: Proceedings of the 2019 ACM international joint conference on pervasive and ubiquitous computing and proceedings of the 2019 ACM international symposium on wearable computers, pp 699–708
52. Al-Halah Z, Rybok L, Stiefelhagen R (2014) What to transfer? High-level semantics in transfer metric learning for action similarity. In: International conference on pattern recognition, pp 2775–2780