Kinect=IMU? Learning MIMO Signal Mappings to Automatically Translate Activity Recognition Systems Across Sensor Modalities

Oresti Baños*, Alberto Calatroni†, Miguel Damas*, Héctor Pomares*,
Ignacio Rojas*, Hesam Sagha‡, José del R. Millán‡,
Gerhard Tröster†, Ricardo Chavarriaga‡, and Daniel Roggen†
**Wearable Computing Laboratory, ETH Zurich, Switzerland*
Email: {alberto.calatroni,daniel.roggen,troester}@ife.ee.ethz.ch
†*CNBI, Center for Neuroprosthetics, École Polytechnique Fédérale de Lausanne, Switzerland*
Email: {ricardo.chavarriaga,jose.millan,hesam.sagha}@epfl.ch
‡*Dep. of Computer Architecture and Computer Technology, University of Granada, Spain*
Email: {oresti,mdamas,hpomares,irojas}@atc.ugr.es

Abstract

We propose a method to automatically translate a preexisting activity recognition system, devised for a source sensor domain S, so that it can operate on a newly discovered target sensor domain T, possibly of different modality. First, we use MIMO system identification techniques to obtain a function that maps the signals of S to T. This mapping is then used to translate the recognition system across the sensor domains. We demonstrate the approach in a 5-class gesture recognition problem translating between a vision-based skeleton tracking system (Kinect), and inertial measurement units (IMUs). An adequate mapping can be learned in as few as a single gesture (3 seconds) in this scenario. The accuracy after Kinect → IMU or IMU → Kinect translation is 4% below the baseline for the same limb. Translating across modalities and also to an adjacent limb yields an accuracy 8% below baseline. We discuss the sources of errors and means for improvement. The approach is independent of the sensor modalities. It supports multimodal activity recognition and more flexible real-world activity recognition system deployments.

1. Introduction

There is a tendency towards an increased availability of sensors readily deployed by users by themselves, with smartphones, sensor-equipped gadgets, smart objects and smart clothing. Living environments are also equipped with more and more sensors for climate control, security, or entertainment. Therefore an important feature of activity recognition systems is to provide a *continuity of context-awareness across different sensing environments*, as the user changes location or carry-on devices. As the user performs their daily activities, various sensor systems will be discovered. These sensor systems may not necessarily be capable of activity recognition, as they may also be deployed for other purposes [1]. For instance, a user relies on a smartphone for activity awareness (e.g. for energy expenditure analysis). S/he enters a room with an activity-aware gaming system and leaves the smartphone on a desk. The smartphone now cannot recognize the user's activities. The gaming system can sense their movements, but may not be devised to recognize the same activities as the smartphone did. Thus, in principle, even if the gaming system sensors deliver relevant data, these data cannot be used to substitute the phone sensors, unless some "translation" occurs.

The question we address is: *how can an activity recognition system devised for one modality (e.g. hand coordinate sensed by a Kinect) be automatically "translated" at runtime to use another modality (e.g. on-body acceleration), and vice-versa?*[1]

1.1. Contributions

- A dataset comprising synchronized 3D coordinates of 15 body joints, measured by a vision-based skeleton tracking system (Microsoft Kinect), and the readings of 5 body-worn inertial measurement units (IMUs): acceleration, rate of turn, magnetic field and orientation. The data is recorded in a 5-class gesture recognition scenario, and for "idle" movements (sec. 3).
- A MIMO system identification technique to learn a function that maps a source signal S (e.g. position) to a target signal T (e.g. acceleration) using data sensed at run-time (sec. 4).
- Two architectures to translate activity recognition systems based on the MIMO mapping (sec. 4):

[1] The Kinect libraries can recognize simple gestures, however we only use the Kinect as a sensor that provides 3D joint coordinates. The results are presented with our own recognition system operating on the Kinect coordinates or on acceleration, and trained in a user-specific manner in both cases. The method to substitute sensor modalities could be equally applied to a user-independent scenario.

1550-4816/12 $26.00 © 2012 IEEE
DOI 10.1109/ISWC.2012.17

1) Kinect → IMU: a system devised for the 3D coordinate of a body joint (wrist) sensed by the Kinect is translated to operate on the 3D acceleration sensed by a wrist-worn IMU.
2) IMU → Kinect: a system devised for the 3D acceleration sensed by a wrist-worn IMU is translated to operate on the 3D coordinate of a body joint (wrist) sensed by the Kinect.

- The fit of the MIMO mapping according to the learning parameters, duration of learning and types of user movements (sec. 5).
- The recognition accuracy after Kinect → IMU and IMU → Kinect translation (sec. 5).
- A discussion of the choice of the translation architecture depending on the nature of the sensors. An analysis of the sources of errors and a discussion of the generality of the method and its potential to tap into rich existing multimedia sources for activity recognition (sec. 6, sec. 7).

2. Related Work

Several approaches improve the substitution of the sensing environment foreseen at design-time by the one effectively encountered at runtime. Sensor-placement-independent activity recognition can be achieved by using datasets collected from multiple on-body locations [2]. This requires training data provided by the user. Self-calibration approaches require no user intervention, but were demonstrated only for specific cases (e.g. displacement of accelerometers [3], [4]). Combinations of multiple sensor modalities also allow to tolerate on-body displacement [5], or to substitute sensor modalities [6]. These combinations must however be predesigned for selected kinds of variations. Alternatively, sensors can self-characterize their on-body placement [7] and orientation [8] to select the appropriate activity models at runtime, but this requires to predefine these models. Transfer learning principles allowing a trained system to transfer activity recognition capabilities to another system have been proposed for body-worn sensors [9] and ambient sensors [10]. These approaches operate on long time scales as they require all the relevant activities to be observed several times (e.g. timescale of days or more). A more exhaustive review and taxonomy of approaches is available in [11]. Overall, these approaches do not fulfill the characteristics desired in this work: they either need to predefine allowed run-time variations, or cannot operate on short time scales, or were not defined for adaptation across sensor modalities.

3. Kinect ↔ IMU Translation Setup

The test bench for this work is a gesture recognition setup (fig. 1) with five body-worn IMUs and a consumer vision-based skeleton tracking system (Microsoft Kinect). These sensors are commonly deployed for activity recognition. IMUs are available on smartphones and can be highly miniaturized. The Kinect allows activity-aware gaming on the XBox console[2]. It has been used for the recognition of activities of daily living [12] and gait analysis [13].

The Kinect contains an 8-bit 640×480 RGB camera, an infrared (IR) LED projecting structured light and an IR camera. It computes on-the-fly an 11-bit 640x480 depth map in a range of 0.7-6m from the reflected IR light. The drivers fit a 15-joint skeleton on the depth map (proprietary algorithm similar to [14]) in real time and deliver 3D joint coordinates in millimeters measured from the Kinect center. Tracking is specified in a range of 1.2-3.5m [15]. The Kinect is interfaced over USB to a PC. We use [16] to record the RGB and depth map videos and the joint coordinates at 30Hz.

Five IMUs (XSens [17]) wired to a PC sense the upper body orientation. We use [18] to acquire the raw sensor data and the device orientation at 30Hz. We only use the 3D acceleration measured by the IMUs.

The Kinect and IMU data are independently recorded and resampled offline to the regular Kinect sampling comb to obtain a synchronized dataset comprising acceleration, position and labels. A single subject performs five kinds of geometric gestures with the right hand in alternation 48 times. These gestures were selected because similar ones can be recognized with wearable sensors [3] or with the Kinect [15]. We also recorded a five minutes long "idle" dataset, where the user performs infrequent low-amplitude arm movements and moves around, without any specific task. The user faces the Kinect within ±30° to avoid occlusions.

4. Translation Method

The translation method works in two steps (fig. 2). First, a system identification technique finds a function that maps the signals of one sensor modality to the signals of another sensor modality. Based on this mapping, the activity recognition system is then translated.

4.1. System Identification (Kinect $\sim$ IMU)

We 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. 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. We denote with $\Psi_{S\to T}$ the function that maps the source to the

[2] The Kinect sold 10 millions units between its release on November 4th, 2010 and March 2011, earning it the Guinness World Record of the "Fastest selling consumer electronic device". Its low cost (150$) puts it in the reach of many households.

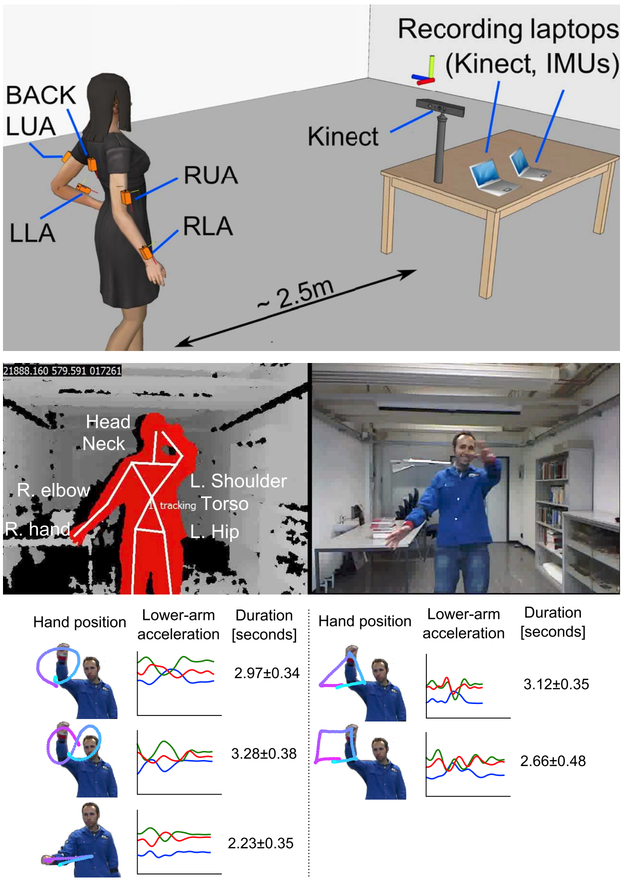


Figure 1: IMUs and a Kinect capture the user's movements (top). The Kinect delivers a depth map, a color image and a 15-joint skeleton of the user (middle). The right hand position and limb acceleration are synchronously recorded for five gesture kinds (bottom).

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

The field of system identification provides techniques to build models[3] of dynamical systems from data [19]. The system identification model should allow for transformations between the typical sensing modalities that are used for activity recognition. Some typical static transformations include scaling (sensors with different sensitivity or units), offset (different zero value), non-linearity (compression of the dynamic range), or rotation. Dynamic transformations may include multiple differentiation or integration operations (e.g. between position or angle and linear or angular velocity), or hysteresis.

Here we use a linear MIMO mapping for system identification [20]. Such mappings can be learned from data. This allows to learn mappings in a wide range of sensing environments without designer involvement or bias. A linear MIMO mapping is defined as follows:

$$\mathbf{x}_\mathbf{T}(t) = \mathbf{B}(l)\mathbf{x}_\mathbf{S}(t) \qquad (1)$$

where $\mathbf{B}(l)$ is the n_T-by-n_S polynomial matrix in the delay operator l^{-1} (i.e. 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 $b_{ik}(l)$ of the form:

$$b_{ik}(l) = b_{ik}^{(0)} l^{-s_{ik}} + b_{ik}^{(1)} l^{-s_{ik}-1} + \ldots + b_{ik}^{(q)} l^{-s_{ik}-q} \qquad (2)$$

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. We identify the $(q+1) \cdot n_T \cdot n_S$ coefficients of the polynomials and the $n_T \cdot n_S$ static delays with a least squares approach.

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.

Specific to this work with Kinect and IMUs:

- We learn a MIMO mapping $\Psi_{K\rightarrow I}$ from the 3D Kinect joint coordinate, to the 3D acceleration (intuitively we see that this requires the MIMO mapping to realize at least a 2nd order differentiation).[4]
- This model can be used both to translate from Kinect to acceleration, and from acceleration to Kinect, thanks to the two translation architectures presented next. The reverse MIMO mapping is not needed.

4.2. Translation Architectures

Two architectures are presented to translate the activity recognition systems. One uses $\Psi_{S\rightarrow T}$, the other uses $\Psi_{T\rightarrow S}$.

[3] Note the distinction between models used for activity recognition ("activity models" in this paper) and models resulting from system identification ("system identification model" or "mapping"). The latter is meant here. It is a mathematical description of the relation between quantities of a physical system, such as the readings delivered by multiple sensors.

[4] We use I or K instead of the S or T subscripts in Ψ or $\mathbf{x}$ to be specific about whether the signals come from the IMUs or the Kinect.

They require the source and target system to exchange either activity models $\mathcal{M}$ or activity templates $\mathcal{T}$.

Template translation architecture (Kinect $\rightarrow$ IMU) The recognition system devised for the source domain also stores the activity templates $\mathcal{T}_S$ that were used for its training. $\mathcal{T}_S$ consists thus of raw sensor signals $\mathbf{x}_S(t)$ and the corresponding class labels. $\Psi_{S\rightarrow T}$ is used to translate the templates $\mathcal{T}_S$ into templates $\mathcal{T}_T$ containing the predicted sensor signals $\hat{\mathbf{x}}_\mathbf{T}(t)$ in the target domain, and the corresponding class labels. System T then runs a feature extraction and selection process, trains a classifier based on $\mathcal{T}_T$, and eventually operates on the data of domain T.

Specific to the Kinect $\rightarrow$ IMU translation, assuming that $\Psi_{K\rightarrow I}$ has been identified:

- The source domain recognition system works on the 3D hand coordinates. It also stores the activity templates $\mathcal{T}_S$ that are the 3D hand coordinates for each gesture.
- $\mathbf{x_S} = \mathbf{x_K}$ is the 3D hand position measured by the Kinect (source)
- $\mathbf{x_T} = \mathbf{x_I}$ is the 3D acceleration measured on the body (target)
- $\hat{\mathbf{x_T}} = \hat{\mathbf{x_I}} = \Psi_{K\rightarrow I}(\mathbf{x_K})$ is the acceleration predicted on the body from the known hand position.
- After template translation, $\mathcal{T}_T$ are the predicted 3D on-body acceleration and the corresponding class labels.
- The target recognition system is automatically trained at run-time on the templates $\mathcal{T}_T$, and finally operates on the acceleration sensed by the IMUs.

Signal translation architecture (IMU $\rightarrow$ Kinect) The recognition system devised for the source domain relies on activity models $\mathcal{M}_S$ (i.e. the parameters of the recognition system, including the selected set of features, the trained classifiers, etc.). After translation, the target recognition system uses the exact same activity models. However, the target system uses $\Psi_{T\rightarrow S}$ to translate the sensor signals of domain T to domain S prior to applying the recognition model $\mathcal{M}_T = \mathcal{M}_S$.

Specific to the IMU $\rightarrow$ Kinect translation, assuming that $\Psi_{K\rightarrow I}$ has been identified:

- The source domain recognition system works on the 3D acceleration sensed by an IMU. It uses models $\mathcal{M}_S$ 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 hand position measured by the Kinect (target)
- $\hat{\mathbf{x_S}} = \hat{\mathbf{x_I}} = \Psi_{K\rightarrow I}(\mathbf{x_K})$ is the acceleration predicted on the body from the hand position.
- After translation, the 3D hand coordinates of the Kinect are mapped to "look like" an acceleration. The recognition models devised for the IMU are used as-is by the target system that now operates on the Kinect data.

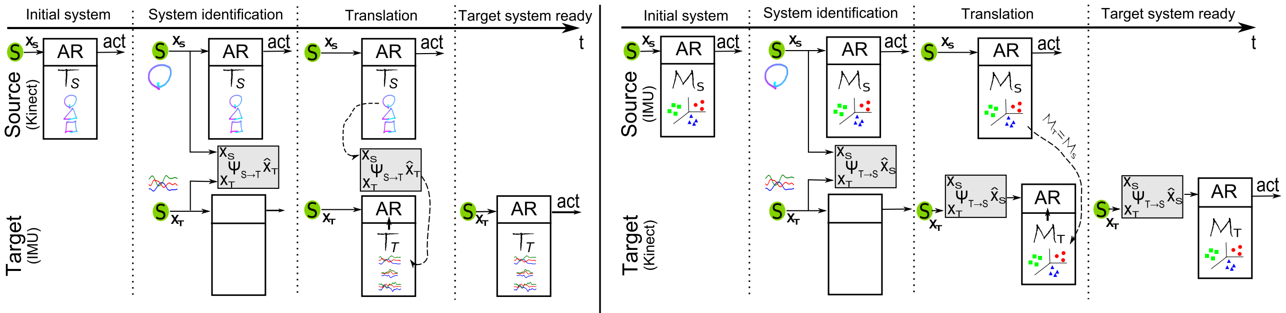
5. Results and Analysis

System identification performance. The translation between Kinect and IMU relies on the identification of $\Psi_{K\rightarrow I}$ which we first characterize. $\Psi_{K\rightarrow I}$ is a 3-input (3D position) 3-output (3D acceleration) MIMO mapping with 10 tap delays (q = 10, 108 parameters to learn). The fit between the measured on-body acceleration $\mathbf{x_T} = \mathbf{x_I}$ and the predicted acceleration $\hat{\mathbf{x_T}} = \hat{\mathbf{x_I}}$, obtained by mapping the source signals (position) $\mathbf{x_K}$ to the target domain is calculated for each channel i:

$$BestFit_i = 1 - \frac{\left(\sum_{t=1}^{N}\left(x_T^{(i)}(t) - \hat{x_T}^{(i)}(t)\right)^2\right)^{\frac{1}{2}}}{\left(\sum_{t=1}^{N}\left(x_T^{(i)}(t) - \bar{x_T}^{(i)}\right)^2\right)^{\frac{1}{2}}} \quad (3)$$

with N the number of signal samples, and $\bar{x_T}^{(i)}$ the mean over time of $x_T^{(i)}(t)$. We average $BestFit_i$ on all channels. A *BestFit* of 1 indicates a perfect fit. Values above zero qualitatively indicate a good fit (fig. 3). The MIMO mappings are learned on a subset of the dataset and evaluated on the rest. The learning subset is obtained by aggregating multiple activity instances, or obtained from the idle dataset. The selection is randomly repeated 20 times in an outer cross-validation process. We evaluate three kinds of MIMO mappings. *Problem-domain mapping (PDM):* This is a generic mapping learned on instances of all classes in equal proportions. *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. Learning is done with a minimum of data corresponding to roughly the duration of a gesture. Thus GSM and UDM are learned on 100 samples and PDM on 500.

The best fit tends to be obtained with PDM (figure 4a). This may be expected, as the mappings are learned on the dynamics of all gestures. Learning can also occur on one gesture of a given class (figure 4b). The best unique gesture to learn a mapping tends to be the circle. Thus, one gesture may be sufficient for the mapping to capture the dynamics of the physical system and extrapolate to a wider range of body movements. UDM does not achieve an adequate mapping (Figure 4c). The idle dataset has only rare occurrences of larger amplitude limb movements and is insufficient to represent the dynamics of the physical system. The fit worsens for mappings between less related body regions. Nevertheless, the fit between hand position and upper arm acceleration is close to that of the hand position to lower arm acceleration. This suggests that translation may be feasible between close-by and related limbs. The back



(a) Template translation architecture

(b) Signal translation architecture

Figure 2: Two architectures allow us to translate a source activity recognition system to operate on a different target sensor domain. The process consists of two steps. First a function $\Psi_{S\to T}$ or $\Psi_{T\to S}$ mapping between the source and target sensor signals is obtained (represented by the gray box). Then the actual translation is performed. AR is an operational recognition system (i.e. trained). It recognizes activities (act) from the data of a sensor (encircled S). a) The source system stores activity templates $\mathcal{T}_S$ that are translated by $\Psi_{S\to T}$ to the target domain $\mathcal{T}_T$; the target system trains its recognition system based on $\mathcal{T}_T$ and is ready to operate. b) The activity models $\mathcal{M}_S$ represent the parameter of the source recognition system; the target system uses these same activity models ($\mathcal{M}_T = \mathcal{M}_S$), and it uses the mapping $\Psi_{T\to S}$ to translate the sensor signals of the target domain to the source domain prior to classification.

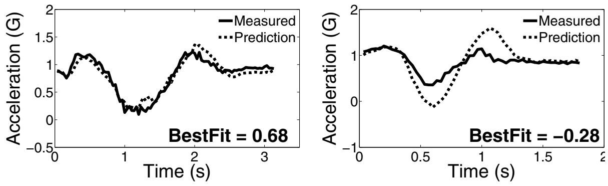


Figure 3: Acceleration at the lower-arm predicted by PDM from the hand position sensed by the Kinect compared to the measured acceleration, for a circle (left) and a slider (right). Visually, a good match between predicted and measured signal is obtained for BestFit values above 0.

acceleration is hardly predictable from the hand position.

Translation accuracy. We characterize the Kinect → IMU translation from the hand position to the acceleration at the lower arm, upper arm, and back, with the template translation architecture. We assess the reverse IMU → Kinect translation to the hand position, from the lower-arm, upper-arm or back acceleration, with the signal translation architecture.

Three non-overlapping parts of the dataset are used to learn the MIMO mapping, to train the source recognition system, and to test the translated target recognition system. The classifier training and testing sets are defined by an instance-based random-seed 5-fold inner cross-validation process, repeated 100 times. The data used to learn the MIMO mapping is selected as indicated previously in an outer cross-validation process. Source and target baseline classification accuracies are assessed by training and testing on data from the same domain (position or acceleration). The Kinect → IMU translation is evaluated by training the recognition system on the predicted acceleration $\hat{\mathbf{x}}_\mathbf{I} = \Psi_{K\to I}(\mathbf{x}_\mathbf{K})$ and testing it on the measured acceleration $\mathbf{x}_\mathbf{I}$. The IMU → Kinect translation is evaluated by training the recognition system on the measured acceleration $\mathbf{x}_\mathbf{I}$ and testing on the predicted acceleration $\hat{\mathbf{x}}_\mathbf{I} = \Psi_{K\to I}(\mathbf{x}_\mathbf{K})$.

Two feature sets are used. Each instance is subdivided into 4 subwindows that capture the temporal dynamics and features are computed on them. FS1 is the mean of each axis (12 features), FS2 is the maximum and minimum of each axis (24 features). We report the accuracy for segmented gestures recognition with a k-NN classifier (k=3).

Classification accuracy baselines in the source (BS) and target domain (BT), and those after transfer to the target domain are presented in fig. 5 for FS2 (this set is used because it is more sensitive to inaccurate signal mapping). The GSM mapping is learned on the “circle” gesture. The baselines indicate that the gestures can be classified with an accuracy of 98% or more with the lower-arm acceleration, the upper-arm acceleration, or the hand position. The high accuracy obtained with the back acceleration (BT of about 88%) indicates that torso movements are 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 performance drop from BS indicates how much worse the system becomes after transfer. The drop from BT indicates how much better would be a system devised specifically for the target domain.

In the translation between hand position and lower or upper arm acceleration, the PDM and GSM models tend to perform equally well. The best results are obtained when translating from hand position to lower-arm acceleration or vice-versa, with less than 4% drop from BS. The drop in performance from BS is less than 8% for the translation from hand position to upper-arm acceleration and vice-versa. The direction of the transfer does not affect the results much. The GSM results show that executing a single "circle" is sufficient to identify a mapping model that leads to a transfer with performance drop between 1% to 7% from BS. The transfer between the hand position and the back acceleration shows a large drop from BS with all mappings (30% to 70%). UDM appear unsuitable for the transfer. This is consistent with the analysis of *BestFit*.

The UDM mapping improves when learned on more "idle" data (fig. 6). With 2000 samples (67 seconds), the performance is about 15% to 30% below the corresponding baselines for FS2 and FS1 respectively. This indicates that, with sufficient data, a dataset from an unrelated domain allows the MIMO mappings to capture the dynamics of the physical system. The difference between FS1 and FS2 highlight that an automatic selection of better features by the source or target system may lead to improved results. Thus, the reported results are a lower bound on the performance.

6. Discussion

Challenges and limitations. Accelerometers measure data in a local frame of reference and the Kinect uses a fixed one. Thus, the signal mapping would have to include not only a second derivative, but also a rotation which is depending nonlinearly 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 (±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.

The Kinect and other video-based tracking systems are affected by occlusions. Since only a small amount of data are needed to learn the mapping, this process is likely feasible in-between occlusions. Furthermore, during an occlusion the *BestFit* decreases, so it may be enough to let the system learn only when *BestFit* is higher than a certain threshold.

Another limitation is that some movements may not be sensed by certain modalities. The Kinect cannot detect torsions of hand and forearm (e.g. in gestures like turning a knob or tightening a screw), but this is easily sensed by gyroscopes and accelerometers, meaning that the expected performance is modality- and gesture-dependent.

Advantages. The approach itself is generic and can be applied to other sensing systems, or to systems of identical modality translated or rotated with respect to each other. The method should scale well with the number of classes, since a mapping learned with one instance of a single class (GSM) performed well on the prediction of the signals of other gestures. This indicates that the mapping approximated the physical relations between the sensing systems, independently of the gestures. We evaluated isolated activity recognition, but the approach is also applicable for continuous recognition (spotting).

Low-variance data unrelated to the activities of interest can be used to learn a mapping (UDM), albeit with more data. 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. Learning, however, benefits from the execution of movements highlighting the physical relation between the sensor systems.

This approach may be useful in crowd-sourcing scenarios [21] to translate generic activity models to the specific sensor modalities that one user has.

The approach was evaluated in simulation. It is however entirely suitable for online, real-time implementation, for instance in sensor nodes.

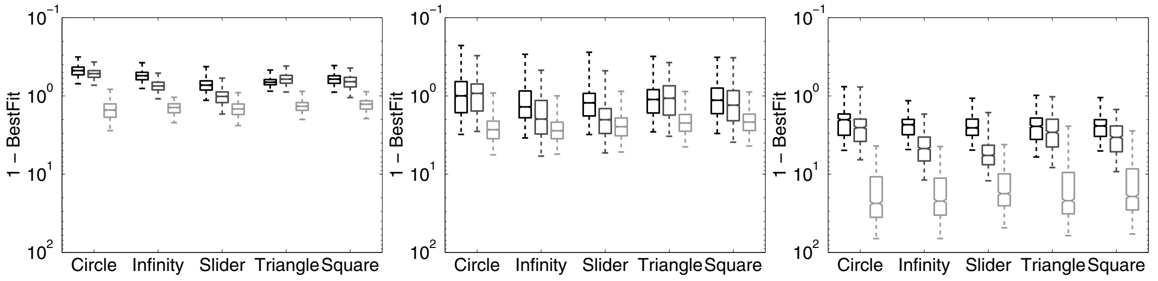
System identification. For specific and tractable cases, a white-box mapping may be devised. Nevertheless, that approach would not generalize to modalities or configurations not foreseen by the system designer. The approach that we propose allows to take advantage of additional sensors as they become available and to learn the mapping without expert intervention.

The approach may be improved by nonlinear or time-varying models, e.g. with time-delay neural networks [22] or nonlinear ARMA [23], or by modeling the transformations between multiple sensors (e.g. two joint coordinates and one acceleration). More complex transformations likely need longer coexistence time between source and target to estimate the model parameters.

BestFit. The *BestFit* indicates the quality of the mapping and to some extent the quality of the resulting translation. This may be an indicator guiding the self-organization of an ecology of sensor systems for opportunistic activity recognition [24].

Since the *BestFit* tends to be highest when sensors measure the movement of a same limb, further investigation may evaluate whether this could be used to automatically localize on-body IMU placement when in range of a skeleton tracking system.

Template translation vs. signal translation. The translation architectures differ in their complexity and memory needs. Template translation does not add computational load on the target system after translation but it requires the source system to store activity templates. This however does not demand large amount of space (47kBytes in floats here). This is well suited for an ambient source and a wearable target system. In contrast, signal translation requires that the target sensor signals are continuously translated. This increases the computational load on the target, but the



(a) PDM

(b) GSM

(c) UDM

Figure 4: Logarithmic box plot 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 the 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.

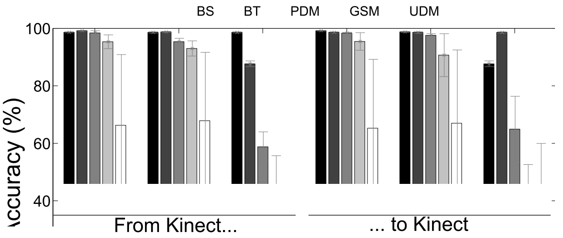


Figure 5: Classification accuracy for the translation between an ambient and wearable system with FS2. Left half: transfer from a system trained on the Kinect hand position to a system operating on the acceleration measured at the indicated positions. Right half: transfer from a system trained on the acceleration signals measured at the indicated positions to a system using the Kinect hand position. BS and BT indicate the baseline accuracies obtained with a system trained and tested on the source and target domain.

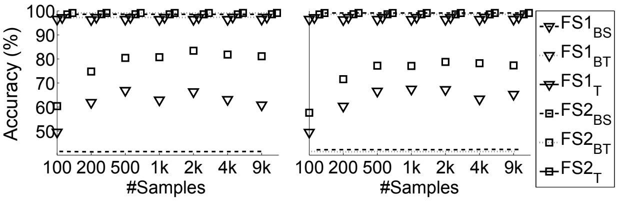


Figure 6: Effect of the amount of idle data used to learn the UDM mapping on the translation accuracy from Kinect hand position to acceleration at the lower arm (left) and vice-versa (right) for features set 1 an 2 (FS1_T, FS2_T). BS, BT are the source and target baselines.

mapping complexity is low and easily benefits from SIMD instructions. 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 architectures also differ in whether the source signal is mapped to the target, or vice versa. If a mapping model exist both ways, then the choice of the architecture is based on computational and memory requirements. If the mapping model is more accurate in one way, then the architecture that uses this mapping is favored. In this work the mapping from position to acceleration influenced the architecture choice.

7. Conclusion

System identification techniques can be used to learn a linear MIMO model that maps 3D positions sensed by a Kinect to the 3D acceleration measured on-body by IMUs.

As few as a single gesture (3 seconds) of data is required to learn the mapping. When the user is idle, more data is required to learn this mapping.

The Kinect $\rightarrow$ IMU and IMU $\rightarrow$ Kinect translation achieves a recognition accuracy of 95%, and is less than 4% below the accuracy of the initial system.

When translating across sensor modalities and also to an adjacent limb (e.g. Kinect hand to IMU on the upper-arm), the accuracy after translation is 8% below baseline.

The approach is generic and could be applied to other sensors, e.g. between a gyroscope and an angle sensor such as a stretch sensor integrated in clothing.

The MIMO models can be replaced by nonlinear ARMA models [23] or time-delay neural networks [22], that may help capture more complex dynamics of the physical system, for instance for combinations of sensors.

This work contributes to activity recognition in open-ended environments. It supports the multi-modal recognition of activities by allowing e.g. to combine video and motion information. In the future, this may be used to learn activity models from existing annotated video sources (e.g. from YouTube), and apply them to movement data sensed on the body (e.g. with a smartphone).

Future work needs to evaluate the approach when frequent occlusions or data loss occur.

Acknowledgment

The project OPPORTUNITY acknowledges the financial support of the Future and Emerging Technologies programme within the 7th Framework Programme for Research of the European Commission, under FET-Open grant number: 225938. This work was also supported in part by the Spanish CICYT Project TIN2007-60587, Junta de Andalucia Projects P07-TIC-02768 and P07-TIC-02906 and the FPU Spanish grant AP2009-2244.

References

[1] A. Calatroni, D. Roggen, and G. Tröster, "A methodology to use unknown new sensors for activity recognition by leveraging sporadic interactions with primitive sensors and behavioral assumptions," in *Proc. of the Opportunistic Ubiquitous Systems Workshop, part of 12th ACM Int. Conf. on Ubiquitous Computing*, 2010.

[2] J. Lester, T. Choudhury, and G. Borriello, "A practical approach to recognizing physical activities," in *Proc. of Pervasive Computing*, 2006, pp. 1–16.

[3] R. Chavarriaga, H. Bayati, and J. del R. Millán, "Unsupervised adaptation for acceleration-based activity recognition: robustness to sensor displacement and rotation," *Pers Ubiquit Comput*, 2012, online first. [Online]. Available: http://dx.doi.org/10.1007/s00779-011-0493-y

[4] K. Förster, D. Roggen, and G. Tröster, "Unsupervised classifier self-calibration through repeated context occurences: Is there robustness against sensor displacement to gain?" in *Proc. 13th IEEE Int. Symposium on Wearable Computers (ISWC)*, 2009, pp. 77–84.

[5] K. Kunze and P. Lukowicz, "Dealing with sensor displacement in motion-based onbody activity recognition systems," *Proc. 10th Int. Conf. on Ubiquitous computing*, Sep 2008.

[6] K. Kunze, G. Bahle, P. Lukowicz, and K. Partridge, "Can magnetic field sensors replace gyroscopes in wearable sensing applications?" in *Proc. 2010 Int. Symp. on Wearable Computers*, 2010.

[7] K. Kunze, P. Lukowicz, H. Junker, and G. Troester, "Where am i: Recognizing on-body positions of wearable sensors," *LOCA'04: International Workshop on Location and Context-Awareness*, Jan 2005.

[8] K. Kunze, P. Lukowicz, K. Partridge, and B. Begole, "Which way am i facing: Inferring horizontal device orientation from an accelerometer signal," in *Proc. of Int. Symp. on Wearable Computers (ISWC)*. IEEE Press, 2009, pp. 149–150.

[9] A. Calatroni, D. Roggen, and G. Tröster, "Automatic transfer of activity recognition capabilities between body-worn motion sensors: Training newcomers to recognize locomotion," in *Proc. 8th Int Conf on Networked Sensing Systems*, 2011.

[10] T. van Kasteren, G. Englebienne, and B. Kröse, "Transferring knowledge of activity recognition across sensor networks," in *Proc. 8th Int. Conf on Pervasive Computing*, 2010, pp. 283–300.

[11] D. Roggen, S. Magnenat, M. Waibel, and G. Tröster, "Wearable computing: Designing and sharing activity-recognition systems across platforms," *IEEE Robotics and Automation Magazine*, vol. 18, no. 2, 2011.

[12] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Human activity detection from rgbd images," in *AAAI Workshop: Plan, Activity, and Intent Recognition*, 2011.

[13] E. Stone and M. Skubic, "Evaluation of an inexpensive depth camera for passive in-home fall risk assessment," in *Proc Pervasive Health Conference*, 2011, pp. 71–77.

[14] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from a single depth image," in *Proc of IEEE Conf on Computer Vision and Pattern Recognition*, 2011, pp. 1297–1304.

[15] "Prime sensor NITE 1.3 algorithms notes, version 1.0," PrimeSense Inc., 2010, http://www.primesense.com.

[16] http://code.google.com/p/qtkinectwrapper/.

[17] *XM-B Technical Documentation*, Xsens Technologies B.V., May 2009, http://www.xsens.com.

[18] D. Bannach, O. Amft, and P. Lukowicz, "Rapid prototyping of activity recognition applications," *IEEE Pervasive Computing*, vol. 7, no. 2, pp. 22–31, 2008.

[19] O. Nelles, *Nonlinear System Identification*. Springer, 2000.

[20] H. Pota, "Mimo systems-transfer function to state-space," *Education, IEEE Transactions on*, vol. 39, no. 1, pp. 97 – 99, feb 1996.

[21] M. Berchtold, M. Budde, D. Gordon, H. Schmidtke, and M. Beigl, "Actiserv: Activity recognition service for mobile phones," in *Proc. 14th Int. Symp. on Wearable Computers (ISWC)*, 2010.

[22] A. Yazdizadeh and K. Khorasani, "Adaptive time delay neural network structures for nonlinear system identification," *Neurocomputing*, vol. 47, no. 1–4, pp. 207–240, 2002.

[23] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[24] D. Roggen, A. Calatroni, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, A. Ferscha, M. Kurz, G. Hölzl, H. Sagha, H. Bayati, J. del R. Millán, and R. Chavarriaga, "Activity recognition in opportunistic sensor environments," *Procedia Computer Science*, vol. 7, pp. 173–174, 2011.