# Who Goes There? Exploiting Silhouettes and Wearable Signals for Subject Identification in Multi-Person Environments

Alessandro Masullo, Tilo Burghardt, Dima Damen, Toby Perrett, Majid Mirmehdi
Department of Computer Science
University of Bristol, Bristol, BS8 1UB, UK
a.masullo@bristol.ac.uk

## Abstract

*The re-identification of people in private environments is a rather complicated task, not only from a technical standpoint but also for the ethical issues connected to it. The lack of a privacy-sensitive technology to monitor specific individuals prevents the uptake of assistive systems, for example in Ambient Assisted Living and health monitoring applications. Our approach adopts a deep learning multimodal framework to match silhouette video clips and accelerometer signals to identify and re-identify the subjects of interest within a multi-person environment. Brief sequences, which may be as short as only 3 seconds, are encoded within a latent space where simple Euclidean distance can be used to discriminate the matching. Identities are only revealed in terms of accelerometer carriers, and the use of silhouettes instead of RGB signals helps to ring-fence privacy concerns. We train our method on the SPHERE Calorie Dataset, for which we show an average area under the ROC curve of 76.3%. We also propose a novel triplet loss for which we demonstrate improving performances and convergence speeds.*

## 1. Introduction

With the recent development of Internet of Things (IoT) technologies, devices employing permanent microphones are an increasingly common sight in people's homes. In spite of this form of monitoring becoming more accepted by users, camera-based home monitoring systems are still far from being as common. In fact, most Ambient Assisted Living (AAL) applications adopt colour images as a main form of input to provide clinically relevant measurements [18]. However, as shown in past research, people's reluctance to install colour cameras in their home, particularly if internet connected, extends to health monitoring applications [26, 28, 1], limiting the use of this powerful technology. In order to reduce the intrusiveness of cameras while *keeping an eye* on patients when staying at home, colour images can be replaced by binary silhouettes. As shown in [15] and [16], silhouettes can be reliably used to monitor patients in their homes and and provide important physiological measurements like the calories burnt and the transition from Sitting-to-Standing (as well as Standing-to-Sitting).

The recent work from the SPHERE project [27] collected data from different homes, including silhouettes, body accelerations and a variety of different environmental sensors' (SPHERE Sensors) data that can be adopted to monitor the overall health of the household. One of the disadvantages of using silhouettes rather than colour images lies in their anonymity. In fact, physiological measurement derived by the analysis of silhouettes would pose quite a challenge to associate with specific individuals, since their silhouettes would be very difficult to distinguish. This is especially true in real-life monitoring applications, where silhouettes can be noisy, low resolution and there is no control over the environment (e.g. presence of guests). This problem is particularly significant in long-term clinically relevant monitoring applications, for example the HEmiSPHERE project [8], where patients undergoing hip and knee replacement spend their rehabilitation period at home while being monitored with the SPHERE Sensors. Being able to isolate the measurements of the monitored participant from the rest of the household is essential for clinicians to investigate the recovery trends of their patient.

In order to solve the re-identification (ReID) problem from silhouettes while preserving the privacy of the member of the household, we provide each monitored participant with a wrist-worn accelerometer. Our deep learning algorithm then maps video and accelerometer streams in a latent space, where the matching between the two can be verified with a distance threshold. Since an accelerometer is body-worn, its measurements are unequivocally assigned to the person being monitored. With this approach, we can compare the silhouettes from every person in the frame with the accelerometer data of each monitored subject to decide whether they belong to the same person or not. Thanks to

our method, anonymous silhouettes can be safely paired to a specific identity using the motion of wearables, therefore maintaining the anonymous aspect of the monitoring process. In addition, the matching of video and accelerometer streams facilitates a wide new range of possibilities for health monitoring applications, allowing for a fusion of multiple sensors that can potentially improve the error of the parameters being monitored. While previous work already attempted the problem of matching videos with accelerometers, they all focused on colour images and required all the participants to wear an accelerometer. In real world applications, patients may have guests, which cannot be required to carry wearables at all times.

In this work, we present a novel multimodal deep-learning detection framework that maps video silhouettes and accelerometer streams into a latent space where a distance threshold can be used to verify the matching. Our novel solution to the Video-Acceleration matching problem can operate on even short ($\approx$ 3 seconds) video snippets, compared to previous methods that require long ($>$ 1 minute) clips to yield satisfactory results. We present results for video-wearable matching on the challenging SPHERE-Calorie dataset [22, 21].

## 2. Related Work

The idea of matching video features with accelerometers to identify subjects in front of a camera has already been explored in the past, and one of the earliest approaches matches trajectories derived from video and accelerometer streams [23]. They suggested a probabilistic approach that maximises the likelihood that subject locations extracted from the cameras correspond with the locations produced by the inertial sensors. Jiang *et al*. [11] used Histogram of Oriented Gradient (HOG) descriptors and a Support Vector Machine (SVM) to generate tracks from colour images of pedestrians, and then compared them with dead-reckoning paths integrated from Inertial Measuring Units (IMU) carried by the recorded subjects. Henschel *et al*. [9] adopted a graph labelling formulation that integrates body worn IMUs and trajectories extracted from a video camera to solve the Video Inertial Multiple People Tracking problem.

While the approach of comparing trajectories to solve the Video-Accelerometer Matching problem works well for outdoor scenarios, it is not suitable for indoor free-living monitoring. In fact, many of the typical indoor activities of daily living do not necessarily require the transition from two different places (e.g. eating, ironing, washing dishes, watching TV...). For these activities, trajectories are less likely to develop, limiting the application of such type of approaches. Moreover, these methods are completely reliant on the performances of the trackers, and IMU based trajectories which are particularly affected by a strong bias that accumulates over time due to the double integration involved in the computations [12].

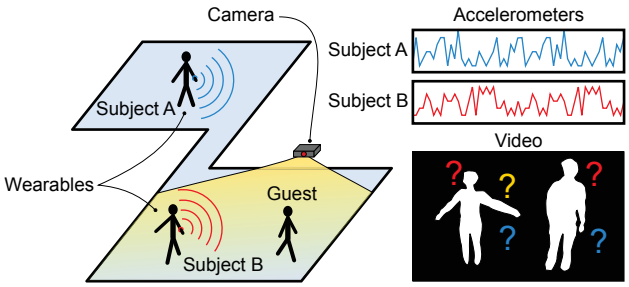



Figure 1. Description of a typical real-life scenario for home monitoring. Two subjects (A and B) are wearing an accelerometer, but only one of them appears within camera view, together with a guest. Our aim is to understand which of the two monitored targets appears in the video silhouette frames.

A different approach to tackle the Video-Accelerometer Matching problem is to estimate accelerations from the video stream. In Shigeta *et al*. [20], a method we shall compare our work against, frames are segmented based on motion and the centroid of each detected area is used to estimate the accelerometer vector. Rofouei *et al*. [17] follow a similar approach using the position of skeleton joints to estimate the acceleration, while Wilson *et al*. [25] estimate the acceleration field using dense optical flows from an infrared camera and convert them into accelerations using depth fields recorded with a Kinect camera and a Kalman filter. All these methods are limited to cases where the wearable device is in the line of sight to the camera and are unable to reliably determine whether the device is in the frame or not.

In [3, 4], another work we shall compare against, Cabrera-Quiros *et al*. tackle the case of crowd-mingling events that include dozens of participants, recorded by cameras, accelerometers and proximity sensors. They estimate acceleration from the video optical flow and use the measurements from proximity sensors to cluster neighbouring people and hierarchically associate them to wearables. A strong limitation of this approach is that every person in the room needs to be carrying the proximity device for the hierarchical method to work. Moreover, their method requires several minutes of recording before being able to reliably match video and accelerometer streams, which may be unsuitable for cases where the subjects frequently move in-between rooms.

The objective of our work is to identify subjects for long-term clinical monitoring using only silhouettes and accelerometers in the shortest time possible. Since we are considering real-life monitoring scenarios, we tackle the rather complex case where only the monitored participants wear an accelerometer whilst being visually recorded amongst other persons who do not wear accelerometers. Our method must be capable of distinguishing people in

such scenarios, as Figure 1 illustrates. In particular, we focus on the challenging problem of matching short segments of video and accelerometer streams, so that quick and clinically relevant movements (*e.g.* Sit-to-Stand [16]) can be associated to a specific individual in spite of the length of the event.

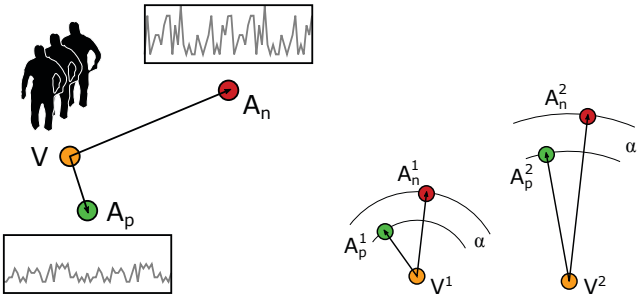
We tackle the Video-Accelerometer Matching problem using a completely new approach, which is inspired by the Active Speaker Detection community. We provided each patient requiring monitoring with a wrist-worn IMU (i.e. a wearable) and we developed an active wearable detection method that identifies which, if any, wearable is active in a specific video sequence. We resolve this problem with a two-stream CNN that encodes both video silhouettes and accelerations in a latent space, where the Euclidean distance is used to discriminate between pairs of matching and non-matching sequences of video and wearable.

## 3. Methodology

Before matching video sequences with accelerometers, the video stream must be processed to detect different subjects appearing in the frame. In our work, we use the person detector and tracker from OpenNI, which provides bounding boxes and tracking information. Similar to Active Speaker Detection works, we developed our framework to match short video/accelerometer clips ($\approx$ 3 seconds). The reason behind this choice is that we are interested in identifying subjects while performing short, clinically relevant movements. As a side-benefit of this choice, we also minimise possible errors of the trackers that, over long periods of time, can mistakenly exchange bounding boxes of different subjects.

### 3.1. Video-Wearable Synchronisation

Let us consider a set of video clips $V = \{V_1, ..., V_N\}$ portraying one person at a time (i.e. the sequence of cropped bounding boxes) while wearing the wristband, and a set of recorded accelerations $A_p = \{A_1, ..., A_N\}$ that constitute a positive match for the videos $V$ by construction. We also define a set of non-matching accelerations $A_n$. The objective of the video-wearable synchronisation is to find two optimal encoding functions $f(\cdot)$ and $g(\cdot)$, so that the Euclidean distance $d$ is minimised for $d\{f(V), g(A_p)\}$ and maximised for $d\{f(V), g(A_n)\}$. The functions $f$ and $g$ are two CNNs that take as input the video clip and the raw accelerations respectively, and produce for output feature vectors. During testing, the synchronisation between a generic video stream and a specific accelerometer can be verified by comparing the Euclidean distance of the two encoded streams against a fixed threshold.




(a) (b)

Figure 2. (a) Example of triplet constituted by an anchor video of silhouettes and two accelerometer sequences for positive and negative matches. (b) Possible problem occurring while training with the Standard Triplet Loss and a fixed margin $\alpha$.

### 3.2. Loss Function

The triplet loss was first proposed to train Siamese Networks for face recognition [19]. A triplet is defined as set of three elements constituted by an anchor, a positive match, and a negative match. In the original work, the triplet was constituted by images of the same person or different faces. In this work, we adapted the triplet using the video as anchor, and a synchronised and non-synchronised sequence of accelerations for the positive and the negative match:

$$(\text{anchor}, \text{positive}, \text{negative}) \equiv (V, A_p, A_n)\,. \tag{1}$$

With this definition of triplet, the loss is defined as:

$$L_{\text{triplet}} = max\{\,|f(V) - g(A_p)|^2 - |f(V) - g(A_n)|^2 + \alpha, 0\}\,, \tag{2}$$

where $\alpha$ is a constant, empirically set to 0.2. The behaviour of the triplet loss is described in Figure 2a: by minimising the quantity described in Eq. (2), the pairs of $(V, A_p)$ are pulled together, while $(V, A_n)$ are pushed apart, to a distance greater than $\alpha$.

In addition to the Standard Triplet Loss (STL), we also experimented using alternative formulations that take advantage of the triplets. One of the problems we experienced with the standard triplet loss is that it does not guarantee that a single threshold can be used to discriminate between matching and not-matching pairs. In fact, the objective of the triplet loss is to separate the $(V, A_p)$ pair from the $(V, A_n)$ pair, no matter what the intra-pair distances are. For example, given two triplets $T^1 \equiv (V^1, A_p^1, A_n^1)$ and $T^2 \equiv (V^2, A_p^2, A_n^2)$ as described in Figure 2b, optimising for the STL ensures that:

$$d\{f(V^1), g(A_n^1)\} - d\{f(V^1), g(A_p^1)\} > \alpha\,, \tag{3}$$

and

$$d\{f(V^2), g(A_n^2)\} - d\{f(V^2), g(A_p^2)\} > \alpha. \tag{4}$$

However, it is entirely possible that the distances are such that $d\{f(V^2), g(A_p^2)\} \gg d\{f(V^1), g(A_n^1)\}$. As it will be shown later, this behaviour is very common for some training strategies and renders the model inoperative, since no single threshold can be used to discriminate between matching and not-matching sequences.

The objective of the training must therefore be such that the model can be used with a single universal threshold. The limitation of the STL is that it becomes identically zero once the distances in Eq. (2) are greater than $\alpha$. To overcome this limitation we implemented a new loss function, named Reciprocal Triplet Loss (RTL), that does not involve any distance margin $\alpha$ and continuously optimises the distances between anchor, positive and negative match. The RTL can be expressed as:

$$L_{\text{RTL}} = |f(V) - g(A_p)|^2 + \frac{1}{|f(V) - g(A_n)|^2} \,. \qquad (5)$$

The characteristic of the proposed RTL is that it is minimised when simultaneously the distances of the good pairs tend to zero and the distances of the bad pairs tends to $+\infty$, therefore maximising the separation between pairs. As shown later in the experiments, the use of the RTL function helps to improve the performance of our model and enables it to operate more robustly with a single universal threshold.

### 3.3. Negative Samples

When the triplet loss is used to train a deep learning model, the samples constituting each triplet must be cleverly selected in a way that they can actively contribute to improving the model. In fact, if the distance between the video anchor and the accelerations from Eq. (2) is greater than $\alpha$, the triplet will have zero loss and it will not contribute to the training. In the original paper on the triplet loss [19], hard mining of triplets was considered as a crucial step to tackle this problem. In our case, the triplets are constrained by the problem of matching videos with accelerometers, and the anchor-positive pair must be a video clip with the synchronised accelerometer sequence. However, the choice of the non-matching acceleration can vary substantially and it has a strong effect on the outcome of the training process.

Let us consider an example where a group of $N$ subjects $(\text{Sub}_1, ..., \text{Sub}_N)$ is performing a set of activities (*standing*, *walking*, *cleaning*, ...). Given an anchor video portraying a subject doing a specific activity, as depicted in Figure 3, a non-matching acceleration can be selected from a different subject doing a different activity (DSDA) or doing the same activity (DSSA), or it could be from the same subject doing the same activity (SSSA) or a different activity (SSDA). The possible combinations of negative samples are summarised in Table 1 for clarity.

| | Same Act. (SA) | Diff Act. (DA) |
|---|---|---|
| Same Sub. (SS) | *SSSA* | *SSDA* |
| Diff. Sub. (DS) | *DSSA* | *DSDA* |
| Overlap | *OVLP* | |

Table 1. Description of possible negative samples for the triplet learning.

The objective of this work is to train a model that learns the synchronisation between video and accelerometer streams. However, if negative samples are only drawn from a different subject doing a different activity (DSDA), the video-wearable matching problem degenerates into a simple activity or identity classifier. Let us consider, for example, a triplet where the anchor is the video of $\text{Sub}_1$ while "walking". The positive match will be the acceleration of $\text{Sub}_1$ while "walking", whereas a DSDA negative could be $\text{Sub}_2$ doing "cleaning", as depicted in Figure 3:

$$\begin{aligned}(V, A_p, A_n) \equiv \big( &\{\text{Sub 1}; \textit{Walking}\}\,, \\ &\{\text{Sub 1}; \textit{Walking}\}\,, \\ &\{\text{Sub 2}; \textit{Cleaning}\}\,\big).\end{aligned} \qquad (6)$$

Since the non-matching acceleration $A_n$ will always be from a different subject doing a different activity, the neural network will try to learn either the identity of the subjects or the activity being performed through the encoding functions $f(\cdot)$ and $g(\cdot)$. Equivalently, training only with DSSA negatives reduces to an activity-agnostic identity classifier, while training with SSDA negatives leads the classifier to only learn activities. A model trained exclusively on DSDA, DSSA or SSDA negatives will not learn anything about the actual synchronisation between the video and the accelerometers, but it will merely compare the action or identity predicted from the video with the one predicted from the accelerometers. This type of model is therefore expected to fail when tested on unseen subjects or activities.

To overcome this limitation and truly associate visual and acceleration features in the temporal domain, a non-matching acceleration can be selected from the same subject while performing the same activity (SSSA). We call this type of negative "hard-negative" (in contrast to the "easy-negatives" DSDA, DSSA and SSDA), since a simple activity or subject classifier is unable to solve this problem and it requires the network to encode information about the actual synchronisation between video and accelerations. In addition to SSSA, a further type of negative sample that we consider is an acceleration that is not synchronised with the video but it is overlapping with it, as presented in Figure 3. We call this type of negative, overlapping (OVLP), and we refer to is as "very hard-negative".

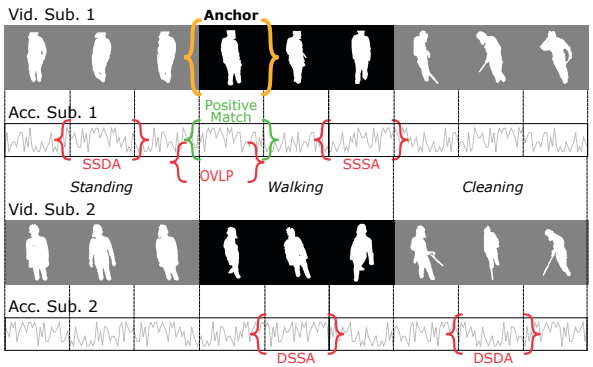



Figure 3. Description of the different possibilities for the negative samples in the triplet. The anchor is the video clip marked in orange, while the positive match is marked in green. A single example of each different negative sample is marked in red.

Table 2. Description of training strategies.

| | Easy | | | Hard | Very Hard |
|---|---|---|---|---|---|
| | *DSDA* | *DSSA* | *SSDA* | *SSSA* | *OVLP* |
| **Easy/Hard** | 25% | 25% | | 50% | |
| **Hard/VeryH** | | | | 50% | 50% |
| **All** | 11% | 11% | 11% | 33% | 33% |

### 3.4. Training strategy

As already highlighted in Section 3.3, different types of negatives can be chosen to form the triplets used to train our model. From an inference point of view, we wish to discriminate between different subjects being monitored while living in their own homes; since the same subject cannot appear in multiple locations at the same time, the validation data only includes negative types of DSDA and DSSA, while the SSSA and SSDA negative types are only used for training. From a training point of view, we already mentioned that a learning strategy using exclusively "easy negatives" would lead the network to become an activity/identity classifier, with potentially poor performance on unseen activities and subjects. In order to learn the actual synchronisation between video and wearables, we tested a variety of training strategies that include different combinations of easy, hard and very-hard negatives, as described Table 2.

The data used in this study (described in detail in Section 4) was split into training and testing based on subject identities, so that the subjects used for testing were never seen during training. Early stopping using this validation data was used to prevent overfitting. Regarding the choice of negative samples, a 50% balance between DSDA and DSSA was chosen and was kept constant across all the experiments.

### 3.5. Data Preprocessing

Each tracked silhouette bounding box in our video data is resized to a constant value and truncated into short clips of 100 frames ($\approx$ 3 seconds) each. In order to avoid any loss of information from the cropping process, bounding box coordinates are also fed into our video encoder network together with the silhouettes. The logic behind this is that the human body can be seen as a deformable body that can either translate or change its shape, and bounding boxes will better capture large rigid displacements (*e.g.* walking) while the cropped silhouettes will address smaller changes within the body shape (*e.g.* wiping a surface).

The accelerometer data is composed of a 3-channel vector, including the IMU measurements in $x$, $y$ and $z$. Typically, machine learning algorithms for audio analysis make use of some transformation of the audio signal in the frequency domain, for example using Perceptual Linear Predictive coefficients (PLPs) [10] or variations of the MFCC [2, 7, 24]. However, since the accelerometer signal is sampled at a frequency that is several orders of magnitude lower than audio (50 Hz for IMU [6] and 32-48 kHz for audio [14]), we decided to feed the raw amplitude of the accelerometers into the network, leveraged by previous works, such as [15], where it was observed that the direct convolution of accelerometer amplitudes yielded satisfactory results. Conceptually, this leaves data transforms to be a responsibility of the network itself. We thereby effectively avoid pre-processing.

### 3.6. Network Architecture

The most important elements of our algorithm are the two encoders $f(\cdot)$ and $g(\cdot)$, which are represented by different CNNs that process the video and accelerometer streams independently to produce the feature vectors. In particular, the video encoder $f(\cdot)$ is the sum of the silhouettes encoder $f_{\text{sil}}(\cdot)$ and the bounding box encoder $f_{\text{bb}}(\cdot)$:

$$f(\cdot) = f_{\text{sil}}(\cdot) + f_{\text{bb}}(\cdot). \tag{7}$$

Our encoders $f_{\text{sil}}(\cdot)$, $f_{\text{bb}}(\cdot)$ and $g(\cdot)$ then constitute a three-stream architecture that is able to take video and accelerometer data in input and produce the distance between the two in the latent space as an output. The architecture for $f_{\text{sil}}(\cdot)$ is presented in Figure 4, with $f_{\text{bb}}(\cdot)$ and $g(\cdot)$ presenting a very similar architecture with 3D operators replaced by their 1D counterpart.

### 3.7. Baseline methods

In order to show the advantages of our method, we implemented two algorithms from the literature to use for baseline comparison. The first is the recent work by Cabrera-Quiros *et al.* [3] where they estimate accelerometer data from the video stream using dense optical flow and then compare it with the actual accelerometer stream. The wearable devices adopted in their experiment also included an embedded proximity sensor that they used to cluster neighbouring devices. Since the target of our study is matching

Table 3. Results of auROC for our Proposed Method trained with the Standard Triplet loss and our novel Reciprocal Triplet Loss, together with the baseline methods.

| | **Standard Triplet Loss** | | | | | **Reciprocal Triplet Loss** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *DSDA* | *DSSA* | *SSSA* | *OVLP* | *AVG* | *DSDA* | *DSSA* | *SSSA* | *OVLP* | *AVG* |
| **Easy/Hard** | 61.2 | 58.6 | 56.9 | 55.4 | **58.0** | 82.9 | 76.8 | 73.6 | 72.0 | **76.3** |
| **Hard/VeryH** | 59.7 | 59.5 | 57.3 | 55.5 | **58.0** | 79.7 | 78.1 | 75.0 | 72.1 | 76.2 |
| **All** | 60.9 | 58.6 | 56.8 | 55.0 | 57.8 | 81.4 | 77.8 | 73.5 | 69.8 | 75.6 |
| | **Baseline** | | | | | | | | | |
| | *DSDA* | *DSSA* | *SSSA* | *OVLP* | *AVG* | | | | | |
| **Shigeta *et al.*** | 56.6 | 56.7 | 56.9 | 57.9 | 57.0 | | | | | |
| **Cabrera-Quiros *et al.*** | 55.8 | 53.2 | 51.0 | 50.6 | 52.6 | | | | | |


Figure 4. Architecture of the video branch $f_{\text{sil}}(\cdot)$ for the three networks tested in this work. The other branches $f_{\text{bb}}(\cdot)$ and $g(\cdot)$ present the same architecture with 3D operators replaced by the 1D counterpart.

video and accelerometer streams without any further sensor input, we implemented their algorithm without the hierarchical approach for the Hungarian method.

In addition to Cabrera-Quiros *et al.*, we also implemented a method inspired from Shigeta *et al.* [20]. In their work, accelerations are estimated using the centroid of each bounding box detected in the video stream and are compared with low-pass filtered version of the accelerometer stream. While implementing this work, our experiments showed that better results were achieved using a high-pass filtered version of the acceleration. Moreover, Shigeta *et al.* use Normalised Cross-Correlation to compare the video and accelerometer signal because their target is streams that are temporally not synchronised. Since we are dealing with a case where the video and accelerometer streams are always synchronised, we compared the signals using Euclidean distance, as per our work.

## 4. Dataset

Our dataset is a modified version of the SPHERE-Calorie dataset [22] which includes RGB-D images, bounding boxes, accelerations and calorie expenditure measures obtained from a Calorimeter, from 10 different individuals doing a set of 11 activities in two different sessions. In this work, we discarded the calorie data and converted the RGB-D images into silhouettes. Silhouettes were generated by processing the RGB images with OpenPose [5] to extract the skeleton joints for each frame of the dataset and then, by running GrabCut on the depth images using a mask initialised with detected skeletons. The dataset included 11 different activities, from which we only kept those actions that involved movement (*i.e.* walking, wiping a surface, vacuuming, sweeping, exercising, stretching, cleaning).

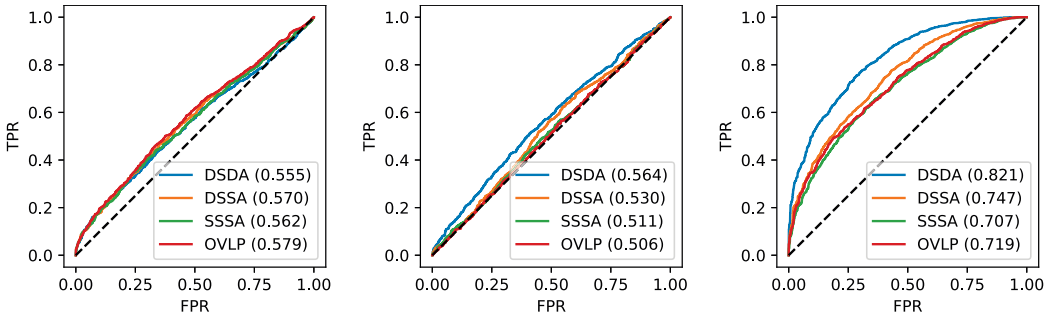
The data from the SPHERE-Calorie dataset was recorded one subject at a time, which enabled us to automatically pair the correct matches between videos and wearables. To simulate the presence of multiple people in each room, we followed the widely adopted strategy of virtual streams [23, 3] whereby the video and accelerometer streams were split into smaller intervals and treated as if they were occurring at the same time. While this approach might be limiting in that subjects never interact with each other, it allows us to push the number of subjects present in a frame beyond the actual capacity of a room, assessing the limits of our method.

## 5. Experiments and Results

We present a series of experiments and ablation tests that are targeted at understanding the advantages and performances of our novel method compared to the state of the art.

### 5.1. Implementation Details

All the networks tested were trained end-to-end using the silhouette video and accelerometer streams in input and the triplet of distances over the embedding in output. The code was implemented using Keras and Tensorflow in Python. Training was performed using the optimiser Adam [13] with a learning rate of $10^{-4}$ and a batch size of 16. We monitored the area under the ROC curves (as later detailed in Section 5.2) after each epoch using the validation data and we stopped training when none of the auROC scores improved for more than 50 epochs. In order to improve performances on the validation data, we implemented some data augmentation strategies. Both the streams of video and accelerometer data were truncated to short clips of $\approx 3$ seconds each using 95% overlap. In addition to that, video silhouettes were randomly flipped (horizontally), dilated (up to 5 pix-




(a) (b) (c)

Figure 5. Receiver Operating Characteristic curves for (a) Shigeta *et al*. [20], (b) Cabrera-Quiros *et al*. [3] and (c) our Proposed Method, computed for 4 different negative types.

els), eroded (up to 5 pixels) and corrupted with salt-and-pepper noise (3%). This strategy, combined with a spatial dropout employed after each convolutional layer, was designed to reduce overfitting of the models on the training data.

### 5.2. Area under the ROC curves

We first evaluate our method on the matching verification task: given a video clip $V_i$ and an acceleration $A_j$, the Euclidean distance between the two embedding $f(V_i)$ and $g(A_j)$ is compared with a threshold $\tau$ to determine the outcome of "matching" or "not matching". While the true matching pairs $P$ are unequivocally defined by the correct pairs of video and accelerometer, the true non-matching $Q$ can be any of the possibilities[1] described in Table 1, resulting in a different score for each negative type. We define the correct true positive matches TP, as a function of the threshold $\tau$, such that:

$$TP(\tau) = \Big\{ (V_i, A_j) \,\Big|\, f(V_i)^2 - g(A_j)^2 < \tau, \\ (V_i, A_j) \in P \Big\} , \quad (8)$$

and the false matches FP as:

$$FP(\tau) = \Big\{ (V_i, A_j) \,\Big|\, f(V_i)^2 - g(A_j)^2 < \tau, \\ (V_i, A_j) \in Q \Big\} , \quad (9)$$

where $P$ and $Q$ are the sets of all positives and all negatives, respectively. By varying the threshold $\tau$, we can plot the true positive rate $TPR$ against the false positive rate $FPR$, defined as:

$$TPR = \frac{TP}{P}, \text{ and } FPR = \frac{FP}{Q} , \quad (10)$$

[1]The reader is reminded that a negative of the "Same Subject" type can never occur in reality, since the same person cannot appear simultaneously in multiple locations. However, we report results for this type of negative because it is useful for our discussion to understand peculiar behaviours of the models trained.

resulting in a ROC curve. The auROC tested with each training strategy (Section 3.4) and the average across negative types (AVG) is presented in Table 3.

Results shows that the best model is achieved by training with a combination of Easy and Hard negatives, employing our novel RTL function. The best model presents an AVG auROC of 76.3%, which constitutes a large improvement over the baseline from Shigeta *et al*. and Cabrera *et al*. who achieve 57.0%s and 52.6%, respectively. In spite of Cabrera *et al*. being designed to deal with crowded events, the lack of the proximity sensor from their implementation and the application to very short clips contribute to the drastic drop in performance in their work. The full ROC curves for the best model are reported in Figure 5, together with the baselines. Once again, the ROC curves confirm the validity of our method and its robustness with a single universal threshold.

If we only consider models trained with the Standard Triplet Loss, the best model only achieves an AVG auROC of 58.0%. Although this result is still better than the baseline, our novel Reciprocal Triplet Loss was found to be essential to guarantee the use of a universal threshold to solve the Video-Accelerometer Matching problem. In addition to that, we also experienced much faster training when using our proposed loss, reaching maximum performances in fewer iterations when compared to the Standard Triplet Loss.

### 5.3. Temporal results

Temporal results for our algorithm are presented in Figure 6 for two example subjects (Subject 10 and Subject 9) from the testing data. We illustrate the situation where both subjects appear in front of the camera but only one of them is wearing a wearable, the other being a guest; the objective is to find which short video clip from each sequence matches the monitored accelerometer. The experiment is even more challenging, since both subjects are simultaneously doing the same sequence of activities. We encoded both the video and accelerometer sequences using the $f(\cdot)$

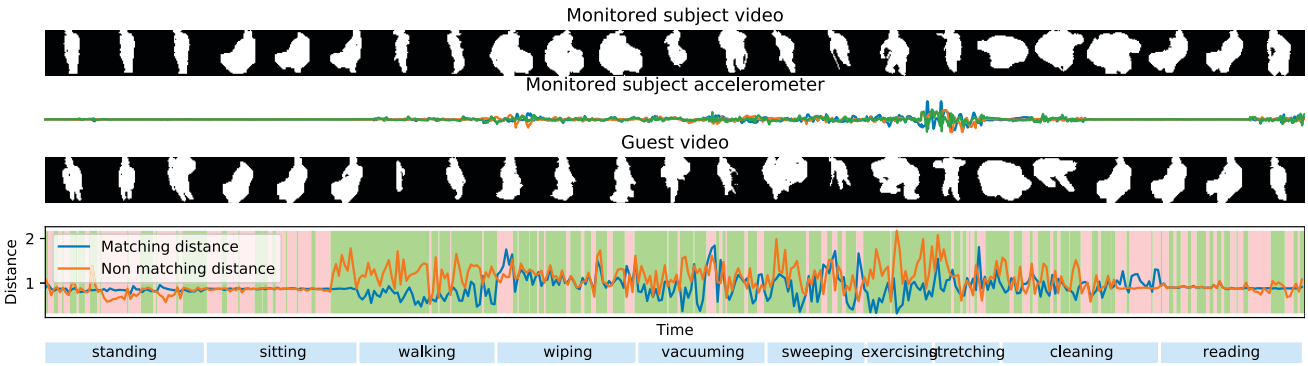



Figure 6. Temporal results for our best model showing the distance between an acceleration sequence and its matching video and the video sequence of a potential guest. The silhouettes shown in the figure are only a subset, sampled for illustration purposes, while the accelerations are presented in $(x, y, z)$ components. The distance plot is highlighted in light green when the matching distance is (correctly) lower than the non matching distance, light red otherwise.

and $g(\cdot)$ deep encoders from the best model we found and we evaluated the Euclidean distances between the two pairs of features:

$$d_{\text{Matching}} = \sqrt{\sum_{i=1}^{N} \left[f\left(\mathrm{V}_9\right) - g\left(\mathrm{A}_9\right)\right]^2}\,, \tag{11}$$

and

$$d_{\text{Non-matching}} = \sqrt{\sum_{i=1}^{N} \left[f\left(\mathrm{V}_9\right) - g\left(\mathrm{A}_{10}\right)\right]^2}\,. \tag{12}$$

The results for this experiment are presented in Figure 6, showing the detailed temporal performances for the best model from Table 3. A very different behaviour can be seen between activities that involve movement (i.e. walking, exercising) and those that do not (i.e. sitting, reading). In fact, active movements involve a variety of gestures that produce a strong motion signature which can be exploited to match video and accelerometers. On the other hand, the output signal of the accelerometers while resting is almost identically nil, no matter which person is wearing it, hindering the ability to match different accelerometers to different video streams.

## 6. Conclusions

Video monitoring for AAL imposes ethical restrictions that can be overcome by using silhouettes instead of colour images. Silhouette anonymity is a double-edged sword that both prevents identification of the household and hinders the ability to identify and track the progress of monitored subjects amongst others. We developed a deep-learning algorithm that encodes short video clips of silhouette and accelerometer streams into a latent space, where the Euclidean distance between the two can be used to discriminate matching and not-matching pairs. We also propose a novel triplet loss function, namely the Reciprocal Triplet Loss, that improves our performances and speeds up the convergence. We demonstrate the validity of our results in a series of experiments and ablation studies, presenting and average auROC of 76.3%. With our results, we show that a deep-learning algorithm largely outperforms traditional methods based on tailored features when tackling the Video-Accelerometer Matching problem. Not only we improved over previous previous results, but we also enabled a solution that allows the use of very short clips down to 3 seconds, compared to several minutes of observation required by previous works. Future work will include further tests on non-scripted datasets and the application to real-life monitoring data for clinical use.

## Acknowledgements

This work was performed under the SPHERE NEXT STEPS project, funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant EP/R005273/1.

## References

[1] G. Birchley, R. Huxtable, M. Murtagh, R. ter Meulen, P. Flach, and R. Gooberman-Hill. Smart homes, private homes? An empirical study of technology researchers' perceptions of ethical issues in developing smart-home health technologies. *BMC Medical Ethics*, 18(1):23, Dec. 2017. 1

[2] H. Bredin. TristouNet: Triplet loss for speaker turn embedding. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5430–5434, Mar. 2017. 5

[3] L. Cabrera-Quiros and H. Hung. Who is where? Matching People in Video to Wearable Acceleration During Crowded Mingling Events. In *ACM on Multimedia Conference*, pages 267–271, New York, New York, USA, Oct. 2016. ACM Press. 2, 5, 6, 7

[4] L. Cabrera-Quiros and H. Hung. A Hierarchical Approach for Associating Body-Worn Sensors to Video Regions in

Crowded Mingling Scenarios. *IEEE Transactions on Multimedia*, 21(7):1867–1879, July 2019. 2

[5] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. 2016. 6

[6] C. Chen, R. Jafari, and N. Kehtarnavaz. A survey of depth and inertial sensor fusion for human action recognition. *Multimedia Tools and Applications*, 76(3):4405–4425, Feb. 2017. 5

[7] J. S. Chung and A. Zisserman. Learning to lip read words by watching videos. *Computer Vision and Image Understanding*, 173:76–85, Aug. 2018. 5

[8] S. Grant, A. W. Blom, M. R. Whitehouse, I. Craddock, A. Judge, E. L. Tonkin, and R. Gooberman-Hill. Using home sensing technology to assess outcome and recovery after hip and knee replacement in the UK: the HEmiSPHERE study protocol. *BMJ Open*, 8(7):e021862, July 2018. 1

[9] R. Henschel, T. V. Marcard, and B. Rosenhahn. Simultaneous Identification and Tracking of Multiple People Using Video and IMUs. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019. 2

[10] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury. Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *IEEE Signal Processing Magazine*, 29(6):82–97, Nov. 2012. 5

[11] W. Jiang and Z. Yin. Combining passive visual cameras and active IMU sensors for persistent pedestrian tracking. *Journal of Visual Communication and Image Representation*, 48:419–431, Oct. 2017. 2

[12] A. Jimenez, F. Seco, C. Prieto, and J. Guevara. A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU. In *IEEE International Symposium on Intelligent Signal Processing*, pages 37–42, Aug. 2009. 2

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] R. Lagadec, D. Pelloni, and D. Weiss. A 2-channel, 16-bit digital sampling frequency converter for professional digital audio. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 7, pages 93–96, 1982. 5

[15] A. Masullo, T. Burghardt, D. Damen, S. Hannuna, V. Ponce-Lopez, and M. Mirmehdi. CaloriNet : From silhouettes to calorie estimation in private environments. *Proceedings of British Machine Vision Conference*, pages 1–14, 2018. 1, 5

[16] A. Masullo, T. Burghardt, T. Perrett, D. Damen, and M. Mirmehdi. Sit-to-Stand Analysis in the Wild Using Silhouettes for Longitudinal Health Monitoring. In *Image Analysis and Recognition*, pages 1–26. Springer Nature Switzerland, 2019. 1, 3

[17] M. Rofouei, A. Wilson, A. Brush, and S. Tansley. Your phone or mine?: fusing body, touch and device sensing for multi-user device-display interaction. In *ACM annual conference on Human Factors in Computing Systems*, page 1915, New York, New York, USA, 2012. ACM Press. 2

[18] S. Sathyanarayana, R. K. Satzoda, S. Sathyanarayana, and S. Thambipillai. Vision-based patient monitoring: a comprehensive review of algorithms and technologies. *Journal of Ambient Intelligence and Humanized Computing*, 9(2):225–251, Apr. 2018. 1

[19] F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 07-12-June, pages 815–823, June 2015. 3, 4

[20] O. Shigeta, S. Kagami, and K. Hashimoto. Identifying a moving object with an accelerometer in a camera view. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3872–3877, Sept. 2008. 2, 6, 7

[21] L. Tao, T. Burghardt, M. Mirmehdi, D. Damen, A. Cooper, S. Hannuna, M. Camplani, A. Paiement, and I. Craddock. Calorie Counter: RGB-Depth Visual Estimation of Energy Expenditure at Home. In *Lecture Notes in Computer Science*, volume 10116 LNCS, pages 239–251. July 2017. 2

[22] L. Tao, A. Paiement, D. Aldamen, S. Hannuna, M. Mirmehdi, I. Craddock, T. Burghardt, and M. Camplani. SPHERE-Calorie. *10.5523/bris.1gt0wgkqgljn21jjgqoq8enprr*, 2016. 2, 6

[23] T. Teixeira, D. Jung, and A. Savvides. Tasking networked CCTV cameras and mobile phones to identify and localize multiple people. In *ACM international conference on Ubiquitous computing*, page 213, New York, New York, USA, 2010. ACM Press. 2, 6

[24] A. Torfi, J. Dawson, and N. M. Nasrabadi. Text-Independent Speaker Verification Using 3D Convolutional Neural Networks. In *IEEE International Conference on Multimedia and Expo*, volume 2018-July, pages 1–6, July 2018. 5

[25] A. D. Wilson and H. Benko. Crossmotion: fusing device and image motion for user identification, tracking and device association. In *International Conference on Multimodal Interaction*, pages 216–223. ACM Press, New York, New York, USA, 2014. 2

[26] W. Zagler, P. Panek, and M. Rauhala. Ambient Assisted Living Systems - The Conflicts between Technology, Acceptance, Ethics and Privacy. *Assisted Living Systems - Models, Architectures and Engineering Approaches*, pages 1–4, 2008. 1

[27] N. Zhu, T. Diethe, M. Camplani, L. Tao, A. Burrows, N. Twomey, D. Kaleshi, M. Mirmehdi, P. Flach, and I. Craddock. Bridging e-Health and the Internet of Things: The SPHERE Project. *IEEE Intelligent Systems*, 30(4):39–46, July 2015. 1

[28] M. Ziefle, C. Rocker, and A. Holzinger. Medical Technology in Smart Homes: Exploring the User's Perspective on Privacy, Intimacy and Trust. In *IEEE Annual Computer Software and Applications Conference Workshops*, pages 410–415, July 2011. 1