Embedded accelerometer signal normalization for cross-device gait recognition

Maria De Marsico,[1] Daniele De Pasquale[2] and Alessio Mecca[3]

Abstract: This paper proposes a "soft" calibration of the signal from smartphone accelerometers, with the aim to improve cross-device gait recognition. Other applications can also benefit from the same procedure. The procedure was evaluated on a dataset of walk signals collected by three different smartphones in two time-separated sessions. The results are extremely satisfactory. For sake of space, only the most significant ones will be reported. In some recognition settings, especially cross-device ones, a relative improvement of over 100% of the starting performance was achieved.

Keywords: cross device gait recognition, smartphone accelerometers

1 Introduction

Gait recognition was formerly mainly carried out by computer vision-based algorithms , but at present a new interest is attracted by the possibilities offered by wearable sensors, e.g., smartphones and the new smartwaches, [Zh15] [DMM15]. Notwithstanding difficulties, using wearable devices is attractive, since it does not require to instrument the environment, and can exploit widespread consumer equipment. Gait features depend on many factors, e.g., the conformation of the lower body, a temporary injury, speed, ground slope, heel height, or carried weights. Further problems can arise from signals themselves due to phase alignment, signal normalization, and denoising. A significant performance decrease is observed when matching signals from different devices, though of the same type, due to systematic errors affecting all physical sensors. These errors are possibly different across the three axes of the accelerometer over which the signal is captured. This can be tolerated for simple tasks on mobile devices, e.g., screen rotation, but biometric recognition needs much higher accuracy. When the use of different devices is required/foreseen, accelerometer data normalization is needed. Physical calibration is not feasible when the accelerometer is built in a smart device. We rather propose a kind of "software calibration strategy". Some available datasets collect data from different devices and from more devices of the same type [Zh15] , but no cross-device matching is attempted. Some works explore inter-device signal acquisition, but experiments usually only assess the correlation among results from different devices of the same model. Even in this case, the obtained signals may present significant differences (see [Gr06], using three activPAL accelerometers). Very few works test matching of signals from different devices of the same type (e.g., [Ma10], comparing activPAL, PALlite and Digi-Walker accelerometers). Most such comparisons deal with activity recognition, and do not attempt correction/avoidance of inter-/cross-device issues. This paper focuses on matching cross-device data from smartphones

[1] Sapienza University of Rome, Information Science, Via Salaria 113, Rome, demarsico@uniroma1.it
[2] Sapienza University of Rome, Information Science, Via Salaria 113, Rome, danieledepasquale@msn.com
[3] Sapienza University of Rome, Information Science, Via Salaria 113, Rome, mecca@uniroma1.it

accelerometers. The very quick normalization procedure proposed does not require hardware intervention or special instrumentation. A new dataset was used to validate it in biometric gait recognition. The contribution of this paper is not a new gait matching algorithm, but rather the proposal and evaluation of an easy, general signal normalization procedure.

2 Related Work

Even the most accurate accelerometer, as any physical sensor, is affected by either systematic or occasional errors. Therefore, even two identical sensors from the same productive cycle provide different measurements in the same conditions. These systematic errors can be highlighted in each single measurement, where the real output differs from the ideal one. An important consideration is that the accelerometer is a linear sensor. Its response is directly proportional to the physical/gravitational acceleration. This allows avoiding complex procedures with special equipment for the device calibration, and rather exploit this property. A relevant parameter for any normalization procedure is $Offset$ (or *Zero-g Offset*, or *Zero-g Bias*). It can be expressed in terms of g (gravitational force, $\tilde{9}.81m/s^2$), and describes the difference of real vs. ideal output when no acceleration is applied to the sensor. When the sensor is on a horizontal flat surface with the frontal part facing up, the ideal value should be $0g$ on X and Y axes and $1g$ on Z axis. Rotating the sensor by $180°$, the values for X and Y axes would remain unchanged while the value for Z would change in $-1g$. Table 1 shows the ideal values. Positions of the sensor are the same shown later in Figure 1 in Section 3, where they rather refer to a smartphone layout. Some existing calibration/normalization procedures inspired our work, that do not use complex equipment, e.g., laser or optical devices. The note in [Tu07] reports four high level proposals to compute the *Zero-g Offset* exploiting the gravity acceleration, and use it to normalize the accelerometer signals. Actually, values of such parameter reported in datasheet are not reliable, especially when the sensor is embedded into another device causing further solicitations. The mentioned procedures are applied directly on the sensor, yet they use high level strategies that do not access, for example, the inner register and/or modify the tension value of the sensor. The first technique exploits *sensitivity* to compute *Zero-g Offset*. The *sensitivity* quantifies the minimum detectable variation. Since the response is linear, this value should be a constant. The procedure obtains it by taking the acceleration at $1g$ ($1g\_value$), and at $-1g$ ($-1g\_value$, obtained by rotating the sensor by $180°$). In these cases only the gravitational acceleration is applied to the sensor. An approximation of *sensitivity* is computed by dividing by 2 the difference between the $1g$ and the $-1g$ values. The *Zero-g Offset* is finally computed as $-1g\_value + sensitivity$ or as $1g\_value - sensitivity$. This procedure is repeated for each axis changing the sensor orientation to have gravity acceleration at $1g$

Tab. 1: The expected values of acceleration when the sensor is laying on a level surface

Positions	Portrait Up	Portrait Left	Portrait Down	Portrait Right	Front	Back
X	0g	+1g	0g	-1g	0g	0g
Y	+1g	0g	-1g	0g	0g	0g
Z	0g	0g	0g	0g	+1g	-1g

in turn for X, Y, and Z. The second technique requires positioning the sensor on a level surface with the frontal face up. It records the acceleration values at $0g$ for the X and the Y axes, and at $1g$ for Z axis. The *Zero-g Offset* for the X and the Y axes is taken as the value recorded at $0g$ while the one for the Z axis is taken as the value recorded for Z at $1g$ minus the *sensitivity*. This procedure does not require repositioning the device, but gives less accurate results and relies on the preliminary knowledge of the sensor *sensitivity*. The third proposal needs registering data at $0g$ in free-falling, at the same time on all three axes. This is seldom feasible for two main reasons. The first one is trivial, if the sensor is embedded in a mobile device. The second is that, during free-falling, the sensor could change its orientation and this causes an erroneous measurement. The last procedure asks putting the sensor on a flat surface facing up, registering the *Zero-g Offset* for X and Y axes, and then repositioning the sensor perpendicular to the surface, to get the *Zero-g Offset* for Z axis. This technique provides poor accuracy due to only 2 measurements per axis.

Being high level is a mandatory requirement for the procedures we propose, because they are targeted at sensors embedded in mobile or wearable devices. Direct operations would not be possible anyway, since they would require extracting the sensor from the mobile device. In addition to this, it is necessary to consider further possible interferences caused by the host device. For these reasons, we had to deeply rethink these inspiring techniques.

3 Accelerometer data normalization

The proposed methodology takes as starting point those in 2. However, it was redesigned to provide reliable results even without direct accelerometer manipulation, when it embedded into a device and cannot be extracted. This required different mathematical computations to increase robustness. The procedure is very simple, can be easily repeated whenever necessary, and the normalization parameters are recorded in a text file stored on the device itself. Therefore they can be used by any applications that may benefit.

The procedure starts by computing the $Offset$ for each axis of the sensor. This is the first difference with related work, where some out of those measures are often only estimated. Then the same normalization procedure is carried out on each single axis. Let us assume for the moment that the $Offset$ (at $0g$) and the value at $1g$, Ref_Value in the following, have been already computed. The general equation 1 synthesizes the normalization formula that we adopt. It is derived from the well-known Min-Max formula, that is used to map a given $Value$ onto a New_Value in $[0,1]$ interval, but represents a variation of this schema. It does not take as reference values the minimum and maximum measured by the accelerometer, that are not easily identifiable for each device, but rather the two reference values measured at $0g$ and $1g$ for each axis (that may fall in different points in the accelerometer range). The general equation is then specialized over the three axes:

$$New\_Value = \frac{Value - Offset}{Ref\_Value - Offset} \tag{1}$$

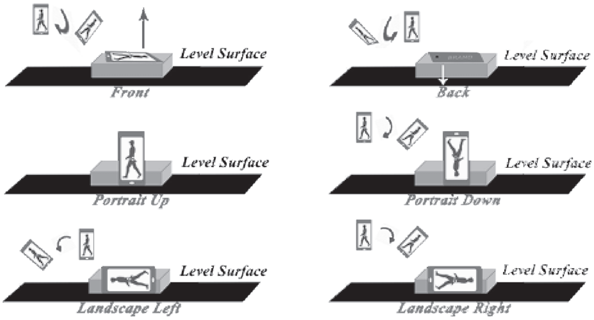
A rescaling in $[0g, 1g]$ is obtained. This allows normalizing the range of values from different accelerometers. Since the accelerometer produces a linear response within the measure range, each movement is translated into a discrete value that is directly proportional to the physical acceleration. By aligning the results with respect to $[0g, 1g]$ it is also possible to achieve a correct alignment of the values originally not included in the same interval.

The goal of the procedure is therefore to compute $Offset$ and Ref_Value for each axis. This requires to carry out two accurate measures per axis. This is another relevant difference with the state-of-the-art methods. Overall, the procedure requires a series of six simple tests. As a further improvement, each single test actually provides a value which is the average of samples taken over a continuous interval of 15 seconds.The device positions to obtain the required measures are shown in Figure 1, where they are referred to the smartphone layout. The detailed procedure steps for one device position are as follows. By setting the device on a plain surface with the screen up (Z at $1g$) it is possible to measure the offset values at $0g$ for X and Y axes and the reference value at $1g$ for Z axis. This is done for 15 seconds, and an average value is computed for each axis. Therefore, from the measurements in this position we obtain three values: Ref_Z_Value and two values that will be used to compute the final offsets for X and Y, i.e., X_Offset_Front and Y_Offset_Front. For each axis, the final value for $Offset$ will be given by the average of 4 values measured at $0g$ in different positions. For instance, as for X axis, such values will be X_Offset_Front, X_Offset_Back, $X\_Offset\_PortraitUp$, and $X\_Offset\_PortraitDown$. Figure 1 also shows the final equations for each axis.

4 The New Collected Dataset and Experimental Results

A new dataset has been collected to test inter-device gait recognition and assess the possible improvements achieved by the proposed normalization. Three smartphones of different brands have been used, each with a different accelerometer model embedded: a OnePlus One (OnePlus - LIS3DH accelerometer by ST Microelectronics), a Samsung Galaxy S4 Active (Samsung - K330 3-axes accelerometer), and a Sony Xperia S (Sony - Bosch Sensortec BMA250 accelerometer). Walk signals belong to 25 subjects, from two acquisition sessions (15 days apart on average). The subjects wore different kinds of shoes but no high heels. Each session includes 6 acquisitions per user, 2 for each smartphone, for a total of 300 walk signals. The subjects were asked to walk normally for ten steps along a straight hallway. After each recording, the smartphone was detached and repositioned to add some further variations. The dataset will be expanded to include more variations due to heels, ground slope and speed, and will be soon freely accessible to the research community [1]. In order to test the advantages of using our normalization procedure, the 5 recognition algorithms proposed in [DMM15] were used. They all exploit the basic formulation of Dynamic Time Warping (DTW), which is still the most used matching method for gait signals. Some of them also use step segmentation. For sake of space, it is not possible to report here the details of the segmentation algorithm. It is worth mentioning that it is based on the identification of signal maxima that fulfill suitable constraints, to avoid the influence of noise. Reporting all results would require too much space, therefore only those for WALK and ALL STEPS VS. ALL are presented below, because they are the most

[1] http://sites.google.com/a/di.uniroma1.it/biometric-interaction/home/gait-recognition/datasets/bwr-multidevice



$$X\_Offset = \frac{X\_Offset\_PortraitUp + X\_Offset\_PortraitDown + X\_Offset\_Front + X\_Offset\_Back}{4}$$

$$Y\_Offset = \frac{Y\_Offset\_LandscapeLeft + Y\_Offset\_LandscapeRight + Y\_Offset\_Front + Y\_Offset\_Back}{4}$$

$$Z\_Offset = \frac{Z\_Offset\_LandscapeLeft + Z\_Offset\_LandscapeRight + Z\_Offset\_PortraitUp + Z\_Offset\_PortraitDown}{4}$$

Fig. 1: The different positions for the offset measurement, and the final equations.

significant in terms of improved performances. WALK algorithm matches unsegmented gait signals. Among those using segmentation, ALL STEPS VS. ALL matches all probe segments against all segments of a gallery walk. Experiments were designed to focus exclusively on normalization effects, by avoiding any denoising and/or interpolation. Aiming at having the most accurate and comprehensive test scenarios, we exploited the 5 recognition algorithms proposed in the mentioned work, and we tested biometric recognition in 3 modalities: Closed Set Identification (CSI), Verification (VER), and Open Set Identification (OSI). Performance were measured in terms of Recognition Rate (RR) for CSI, and in terms of Equal Error Rate (ERR) for VER and OSI. Half of these tests were performed using the walk signals without normalization and the other half using the same data after the application of the proposed normalization procedure. The full set of analyzed test scenarios (*ts*s) was created combining the following conditions:
AllSessions = probe and gallery sets belong to both sessions.
Session_vs_Session = one session in turn is used as probe set and the other as gallery set.
SameSession = probe and gallery sets belong to the same session.
AllDevices = both probe and gallery sets belong to all devices.
Device_vs_Device = one device is used in turn as probe and one as gallery source.
SameDevice = both probe and gallery set come from the same device.
The only missing combination is SameSession with Device_vs_Device, because it is not much realistic. Summing up there is a total of 38 *ts*s, each one enacted with each of the 5 recognition algorithm in each of the 3 different biometric modalities, for a total of 570 tests. Moreover, each test was repeated with and without normalization, for a total of 1140 tests, out of which we report those carried out with the selected recognition algorithms.

Tab. 2: Results with WALK. The green cells (lighter) report improvement while the red ones (darker) decrements. RR=Recognition Rate, EER=Equal Error Rate, O.D.=Original Dataset, N.D.=Normalized Dataset, AD=AllDevices, D_vs_D=Device_vs_Device, SD=SameDevice, AS=AllSessions, S_vs_S=Session_vs_Session, SS=SameSession, OP=OnePlus, Sams=Samsung.

Recognition Method: WALK												
Test Scenario		Device		Closed Set Identification			Verification			Open Set Identification		
				RR O.D.	RR N.D.	Improv.	EER O.D.	EER N.D.	Improv.	EER O.D.	EER N.D.	Improv.
AS	AD			95.00%	97.00%	2.11%	28.00%	24.10%	16.18%	26.30%	23.70%	10.97%
	D_vs_D	OP	Sams	90.00%	92.00%	2.22%	20.70%	21.00%	-1.45%	38.00%	35.00%	8.57%
		OP	Sony	65.00%	87.00%	33.85%	29.50%	24.70%	19.43%	58.00%	43.00%	34.88%
		Sony	Sams	74.00%	88.00%	18.92%	27.30%	24.80%	10.08%	53.00%	35.50%	49.30%
		Sony	OP	76.00%	87.00%	14.47%	29.50%	24.70%	19.43%	57.00%	39.00%	46.15%
		Sams	OP	97.00%	99.00%	2.06%	20.70%	21.00%	-1.45%	30.00%	25.50%	17.65%
		Sams	Sony	70.00%	93.00%	32.86%	27.30%	24.80%	10.08%	56.00%	34.00%	64.71%
	SD	OP		94.00%	94.00%	0.00%	24.60%	23.50%	4.68%	24.60%	23.60%	4.24%
		Sony		90.00%	90.00%	0.00%	28.00%	28.00%	0.00%	28.00%	28.40%	-1.43%
		Sams		96.00%	96.00%	0.00%	24.40%	24.00%	1.67%	24.40%	24.00%	1.67%
S_vs_S	AD			52.00%	54.50%	4.81%	31.80%	29.60%	7.43%	31.80%	29.60%	7.43%
	D_vs_D	OP	Sams	39.00%	49.00%	25.64%	28.08%	28.45%	-1.34%	75.00%	70.00%	7.14%
		OP	Sony	32.00%	48.00%	50.00%	32.45%	26.43%	22.80%	82.00%	73.00%	12.33%
		Sony	Sams	27.00%	46.00%	70.37%	33.50%	31.55%	6.18%	82.00%	78.00%	5.13%
		Sony	OP	27.00%	42.00%	55.56%	32.50%	31.75%	2.36%	85.00%	76.00%	11.84%
		Sams	OP	50.00%	55.00%	10.00%	28.08%	30.53%	-8.73%	70.00%	70.00%	0.00%
		Sams	Sony	37.00%	56.00%	51.35%	33.50%	28.38%	18.06%	81.00%	67.00%	20.90%
	SD	OP		56.00%	57.00%	1.79%	27.00%	26.80%	0.75%	27.00%	26.80%	0.75%
		Sony		43.00%	45.00%	4.65%	31.20%	31.20%	0.00%	31.20%	31.20%	0.00%
		Sams		52.00%	54.00%	3.85%	28.20%	29.00%	-2.84%	28.20%	29.00%	-2.84%
SS	AD			94.50%	95.50%	1.06%	21.40%	13.35%	60.30%	23.00%	22.10%	4.07%
	SD	OP		91.00%	91.00%	0.00%	8.00%	8.00%	0.00%	25.00%	25.00%	0.00%
		Sony		94.00%	95.00%	1.06%	6.00%	6.00%	0.00%	20.00%	20.00%	0.00%
		Sams		72.00%	72.00%	0.00%	16.00%	15.95%	0.31%	30.50%	30.00%	1.67%

An all-against-all matching was always carried out, where, given the chosen scenario, each template in the probe set is used in turn and compared with all the gallery.

All tests in all scenarios demonstrate a generally significant performance improvement, that arrives to 225%. Even for tests involving a single device there is an extremely high improvement, though unexpected given that they involve exactly the same smartphone. Analyzing the best results only among those reported in Table 2 for WALK matching algorithm and in Table 3 for ALL STEPS VS. ALL, we observe that in *ts* Same_Device combined with AllSessions with ALL STEPS VS. ALL, we start from confirming that, also using a single device, normalization increases matching accuracy. In fact, we got up to a 89.74% relative improvement in CSI, achieved using Samsung with which the results pass from 39% to 74% of RR; the improvement in VER reaches 157.14%, with the performances passing from 36% to 14% of EER, and a 75.68% improvement in OSI, with performances passing from 70% to 30% of EER, in the case of Same_Device combined with SameSession using OnePlus with ALL STEPS VS ALL.

In the cases of Device_vs_Device *ts*s, we got the following improvements: up to 131.25% in CSI, when combined with AllSessions using templates from Samsung as probes and

Tab. 3: Results with ALL STEPS VS. ALL. The green cells (lighter) report improvement while the red ones (darker) decrements. RR=Recognition Rate, EER=Equal Error Rate, O.D.=Original Dataset, N.D.=Normalized Dataset, AD=AllDevices, D_vs_D=Device_vs_Device, SD=SameDevice, AS=AllSessions, S_vs_S=Session_vs_Session, SS=SameSession, OP=OnePlus, Sams=Samsung.

Recognition Method: ALL STEPS VS. ALL												
Test Scenario		Device		Closed Set Identification			Verification			Open Set Identification		
				RR O.D.	RR N.D.	Improv.	EER O.D.	EER N.D.	Improv.	EER O.D.	EER N.D.	Improv.
AS	AD			51.00%	78.00%	52.94%	50.00%	50.00%	0.00%	57.70%	38.60%	49.48%
	D_vs_D	OP	Sams	45.00%	67.00%	48.89%	50.00%	50.00%	0.00%	70.50%	48.50%	45.36%
		OP	Sony	32.00%	52.00%	62.50%	44.00%	37.70%	16.71%	79.00%	61.50%	28.46%
		Sony	Sams	28.00%	57.00%	103.57%	50.00%	50.00%	0.00%	84.00%	57.00%	47.37%
		Sony	OP	38.00%	56.00%	47.37%	44.80%	33.30%	34.53%	74.00%	58.50%	26.50%
		Sams	OP	54.00%	76.00%	40.74%	50.00%	50.00%	0.00%	67.00%	48.00%	39.58%
		Sams	Sony	32.00%	74.00%	131.25%	50.00%	50.00%	0.00%	67.00%	48.00%	39.58%
	SD	OP		43.00%	73.00%	69.77%	40.30%	32.70%	23.24%	40.30%	32.60%	23.62%
		Sony		52.00%	51.00%	-1.96%	40.30%	38.80%	3.87%	40.30%	38.80%	3.87%
		Sams		39.00%	74.00%	89.74%	50.00%	50.00%	0.00%	50.00%	50.00%	0.00%
S_vs_S	AD			22.50%	34.50%	53.33%	50.00%	50.00%	0.00%	83.30%	71.30%	16.83%
	D_vs_D	OP	Sams	17.00%	25.00%	47.06%	45.00%	38.95%	15.53%	92.50%	85.00%	8.82%
		OP	Sony	15.00%	31.00%	106.67%	44.60%	38.05%	17.21%	90.00%	83.50%	7.78%
		Sony	Sams	13.00%	23.00%	76.92%	47.50%	39.65%	19.80%	94.50%	91.00%	3.85%
		Sony	OP	14.00%	22.00%	57.14%	47.65%	46.10%	3.36%	92.00%	91.50%	0.55%
		Sams	OP	16.00%	30.00%	87.50%	50.00%	50.00%	0.00%	93.50%	91.00%	2.75%
		Sams	Sony	16.00%	29.00%	81.25%	48.50%	42.05%	15.34%	90.50%	90.00%	0.56%
	SD	OP		18.00%	34.00%	88.89%	42.85%	36.55%	17.24%	90.00%	80.00%	12.50%
		Sony		25.00%	22.00%	-13.64%	41.85%	42.75%	-2.15%	84.00%	88.00%	-4.76%
		Sams		23.00%	31.00%	34.78%	50.00%	49.90%	0.20%	89.50%	82.00%	9.15%
SS	AD			54.00%	79.00%	46.30%	44.05%	35.00%	25.86%	56.50%	38.35%	47.33%
	SD	OP		40.00%	71.00%	77.50%	36.00%	14.00%	157.14%	65.00%	37.00%	75.68%
		Sony		54.00%	54.00%	0.00%	27.60%	26.25%	5.14%	58.50%	57.50%	1.74%
		Sams		40.00%	75.00%	87.50%	40.40%	32.00%	26.25%	74.00%	48.50%	52.58%

templates from Sony as gallery, with the ALL STEPS VS. ALL, with the result increasing from 32% to 74% of RR; we got up to 34.5% improvement in VER when combined with AllSessions using Sony templates as probes and OnePlus templates as gallery with the ALL STEPS VS. ALL, with results improving from 44.8% to 33.3% of EER; finally, we got up to 64.71% relative improvement in OSI when combined with AllSessions using Samsung templates as probes, and Sony templates as gallery with the WALK recognition method, with results passing from 56% to 34% of EER. Notwithstanding the peaks due to worse starting values, the improvements in cross-device matching are higher on average than those obtained when using the same device.

Table 2 shows the complete set of results for the *ts*s with the recognition method WALK, that experimentally achieved the best results among the recognition methods in [DMM15], even without normalization. For symmetric conditions, e.g. the pair of Session_vs_Session, we just report the average results. With WALK, the global performances stay the same or increase. The *ts*s involving different devices, especially in Device_vs_Device setting, always achieve an improvement in all three biometrics testing modalities (except for few *ts*s in VER and in OSI). The improvements are up to 70.37% for CSI, up to 60,3% for VER, and up to 64,7% in OSI. It is worth pointing out that the major benefits of normalization, as expected, are in Device_vs_Device *ts*s.

Table 3 shows the complete set of results for *ts*s with the recognition method ALL STEPS VS. ALL, that is the one that achieved the best benefit from normalization. Even in this case, for symmetric conditions we just report the average results. The global performances are lower in this case. This matching algorithm is free from the limitation to require about the same number of steps/segments in the signals to match, and therefore this result was expected due to possible greater inaccuracy. In fact, this algorithm uses step segmentation, and matches single steps having a limited number of signal points (about 100). Therefore it is more affected by signal distortions, such as systematic errors. In fact, in the not normalized dataset, it achieves quite poor performances, yet it is interesting to notice that normalization provides an even higher improvement, especially in Device_vs_Device *ts*s, so that we achieve a twofold important result.

5 Conclusion and Future Work

Solutions to improve accelerometer signal quality for gait recognition, at the best of our knowledge, do not tackle extensively cross-device signal matching. This paper has proposed an effective procedure for signal normalization. It was not possible to test it on existing datasets, due to the lack of the required measures, but it can be easily implemented for a brand new system gallery, either during acquisition, or even afterward, given that the requested measures are computed later. Experimental results demonstrate that normalization also positively affects matching of data from the same device, though being especially beneficial in cross-device matching. This reveals that it is not possible to export normalization parameters from one sensor to another, though of the same model.

References

[DMM15] De Marsico, Maria; Mecca, Alessio: Biometric Walk Recognizer. In: New Trends in Image Analysis and Processing–ICIAP 2015 Workshops. Springer, pp. 19–26, 2015.

[Gr06] Grant, P Margaret; Ryan, Cormac G; Tigbe, William W; Granat, Malcolm H: The validation of a novel activity monitor in the measurement of posture and motion during everyday activities. British journal of sports medicine, 40(12):992–997, 2006.

[Ma10] Maddocks, Matthew; Petrou, Andrea; Skipper, Lindsay; Wilcock, Andrew: Validity of three accelerometers during treadmill walking and motor vehicle travel. British journal of sports medicine, 44(8):606–608, 2010.

[Tu07] Tuck, Kimberly: Implementing auto-zero calibration technique for accelerometers - Application Note AN3447. Freescale Semiconductor Application Note, 2007. [AccessedApril 2016].

[Zh15] Zhang, Yuting; Pan, Gang; Jia, Kui; Lu, Minlong; Wang, Yueming; Wu, Zhaohui: Accelerometer-based gait recognition by sparse representation of signature points with clusters. Cybernetics, IEEE Transactions on, 45(9):1864–1875, 2015.