Comparing Motion Data from an iPod Touch to an Optical Infrared Marker-Based Motion Capture System

Kristian Nymoen,[1] Arve Voldsund,[1,2] Ståle A. Skogstad,[1]
Alexander Refsum Jensenius,[2] and Jim Torresen[1]
[1]Department of Informatics, University of Oslo, Norway
[2]Department of Musicology, University of Oslo, Norway
{krisny,savskogs,jimtoer}@ifi.uio.no,{arve.voldsund,a.r.jensenius}@imv.uio.no

ABSTRACT

The paper presents an analysis of the quality of motion data from an iPod Touch (4th gen.). Acceleration and orientation data derived from internal sensors of an iPod is compared to data from a high end optical infrared marker-based motion capture system (Qualisys) in terms of latency, jitter, accuracy and precision. We identify some rotational drift in the iPod, and some time lag between the two systems. Still, the iPod motion data is quite reliable, especially for describing relative motion over a short period of time.

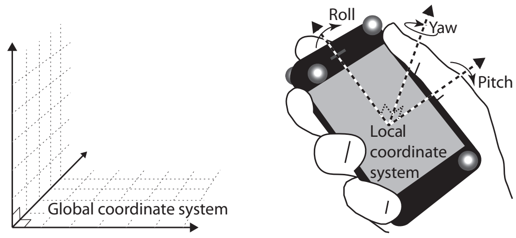


Figure 1: The iPod, defined as a rigid body, enables Qualisys tracking of orientation and position.

1. INTRODUCTION

With advances in mobile technology during the last years, mobile devices have become increasingly popular for musical interaction. In this paper we will focus on Apple's iOS devices, which come with a variety of sensors, depending on the type and model: touch screen, accelerometer, gyroscope, GPS, and magnetometer. Additionally, pre-processed data extracted from the raw sensor data, e.g. orientation and acceleration, is made available through the iOS SDK.

The motivation for the present study is to learn more about the quality of the motion data from an iPod Touch. Several researchers have reported on strengths and weaknesses of iOS devices, e.g. [9, 11], but, these are rarely quantified. In order to know how precisely a motion feature can be reproduced, how fast an action can be recognized, and so forth, we need quantitative evaluations of the data.

Some musical parameters may require high precision and accuracy, while other parameters do not, and with the proper knowledge about the quality of the iPod data, we can make more qualified decisions when mapping motion parameters to musical parameters. This paper evaluates data from an iPod Touch by comparing it to data from a state-of-the-art optical marker-based motion capture (mocap) system from Qualisys, through analyses of timing (i.e. latency and jitter), as well as accuracy and precision, hereunder drift and noise of orientation and acceleration data.

2. BACKGROUND

In the last decade or so, we have seen an increased interest of mobile phones for musical applications in the NIME community and elsewhere. PDAs [18] and Nokia phones [7] have been used, in addition to the increasing number of applications developed for iOS devices in the last years, e.g. [4, 12, 19]. Recently, mobile devices have also become the main computing platform of certain formalised ensembles, e.g. [14].

Several general purpose environments for working with music on mobile phones have been developed, including versions of PureData for PDAs [10], mobile phones [15], and the *libpd* port of PureData to iOS and Android [2]. Moreover, the Synthesis ToolKit (STK) has been ported to Symbian [6], and iOS [3], and the urMus environment has been designed for rapid design of mobile musical instruments [5].

In [8], Essl and Rohs present a design space based on using sensors on mobile devices for developing musical instruments. They emphasise the importance of considering the properties of the sensors at hand. Specifically for gyroscopes and accelerometers, which are used in the iPod Touch discussed in the present paper, they mention that these sensors are good for measuring relative motion, but that the lack of an absolute frame of reference makes absolute measurements difficult. Through the experiments presented in the next chapter, we have aimed to quantify such measures.

3. EXPERIMENT

We have used data from a Qualisys optical infrared marker-based mocap system as a reference when evaluating the iPod data. Our setup consisted of 9 Oqus 300 cameras, operating at a sampling rate of 150 Hz. The system is reported to have a high spatial resolution. However, this resolution depends on the distance between the object that is being captured and the mocap cameras, in addition to the calibration quality [17].

The iPod (Figure 1) was equipped with four reflective markers ($\varnothing$ = 12 mm). The configuration of the markers was used to define the iPod as a rigid object, with centre position at the geometric centre of the markers. In this manner, we used the optical motion capture system to record the position and the orientation of the iPod.

3.1 iPod

We used an iPod Touch, 4th generation, running iOS version 4.3.5, for the experiment. The device contains a three-

NIME'12, May 21 – 23, 2012, University of Michigan, Ann Arbor.

axis accelerometer and gyroscope, which is used to calculate certain motion features on the iPod. We have not used the raw data values from the accelerometer and gyroscope, but rather utilised the motion features that are available through the `CMDeviceMotion` class in the iOS Developer library [1]: *attitude*, *rotationRate*, *gravity*, and *userAcceleration*. The reason for using these features is that they are intended to be conceptually similar to the data provided by the Qualisys system, as opposed to what raw sensor data (e.g. from an accelerometer) would be.

We have developed an application for accessing these data from the iPod. The motion features were sampled at 60 Hz, and packed into OpenSound Control (OSC) bundles. These were sent via UDP over a wifi network set up by the recording computer. The 60 Hz sampling rate was set in the iPod application at the development stage, and other sampling rates have not been tried in this paper.

3.2 Recordings

The data from the Qualisys system was sent as OSC bundles via UDP over Ethernet to the recording computer. The iPod data and Qualisys data were recorded in Max5, as separate streams in an SDIF file, to obtain synchronized recordings of the motion data [13]. The recorded data types are presented in Table 1, these were provided natively from the devices. In this table, *global* means that the data stream is given in relation to some global, fixed, coordinate system, and *local* means that the data stream is measured in relation to the local coordinate system of the iPod (Figure 1).

The iPod was held in one hand, and a total of 22 recordings were made. These included short recordings of tilting the iPod around each of the rotational axes individually, as well as longer, spontaneous rotational and shaking gestures (durations ≈ 4–23 seconds). Furthermore, a ten minute recording was made where the iPod was lying still. Orientation was recorded both as Euler angles and 3×3 Direction Cosine Matrix (DCM).[1] Since the coordinate systems from the iPod and Qualisys were not aligned, the iPod orientation data was adjusted by hand to match the Qualisys data during postprocessing.

Table 1: Recorded motion data

Qualisys		iPod	
Orientation	Global	Orientation	Global
Position	Global	User Acceleration	Local
Marker pos.	Global	Gravity	Local
		Rotation rate	Local

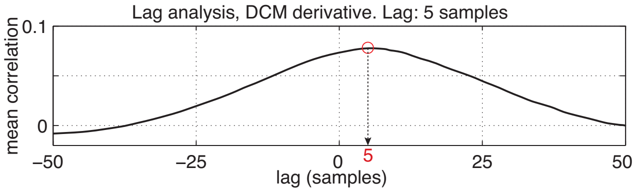
4. ANALYSIS

We start the data analysis by looking at issues related to timing, including latency and jitter. Subsequently, we move on to accuracy and precision of rotational and positional data. For the analysis, we selected a subset of the recordings where there were no gaps in the motion capture data (i.e. the rigid body was tracked at every frame). The results presented in this section are discussed in Section 5.

4.1 Timing

4.1.1 Lag

We observed a time lag between the data from Qualisys and the iPod. To analyse this, we performed cross-correlation on the derivatives of the DCM elements, for eight recordings. Cross-correlation measures the similarity between the two data streams as a function of a time lag applied to one of the streams. Using the derivatives removes inaccurately high correlation scores of stationary extreme-value elements. To achieve an equal number of samples in the data streams, the iPod data was up-sampled to 150 Hz using cubic spline interpolation before the derivative was calculated. By averaging the cross correlations, we achieved an estimate of the time lag between Qualisys and the iPod for each recording, as shown for one recording in Figure 2, the figure also shows that for the eight recordings, the mean lag between Qualisys and iPod data was 43 ms, standard deviation (SD) 8 ms.



	mean	SD	min	max
Lag (ms)	43	8.0	33	53

Figure 2: The plot shows the averaged cross-correlation between the DCM elements of the iPod versus Qualisys for one recording. In this recording, the lag is 5 samples (∼33 ms). The table below shows lag statistics for 8 recordings. Qualisys and iPod correlation is highest when shifted by 43 ms.

4.1.2 Jitter

For applications where the iPod sensor data is sent to an external device, it can be crucial that the timing of received data packets is stable. To evaluate the temporal stability of the system, we measure jitter, as the variation in the time interval between received OSC bundles, in a sequence of 1000 samples. Figure 3 shows a histogram and statistics of the time intervals between successive samples. The standard deviations give indications of the amount of jitter in the data streams. This measure is high for both systems, suggesting that variations in the network connections between the sensing devices and the receiving computer might be partly responsible for this. Still, the standard deviation is notably higher for the iPod than for the Qualisys system, suggesting that the iPod is less reliable when it comes to delivering data packets at regular time intervals.

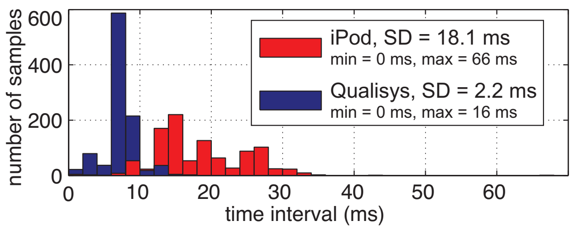


Figure 3: Histogram of the time interval between 1000 successive received samples.

4.2 Accuracy and Precision

4.2.1 Orientation Data

It has been shown that Spearman's rank correlation is suitable for comparing data with serial correlation [16]. We applied this to the 9 DCM elements of the iPod and Qualisys to analyse accuracy of the orientation data. Again, a cubic spline was used to upsample the iPod data, and the data was time-shifted and trimmed according to the iPod lag, as described in Section 4.1.1.

Figure 4 shows a histogram of the correlation coefficients for the 9 DCM elements for 8 recordings. 2/3 of the correlation coefficients are above 0.96, which indicates that in

[1] Rotation Matrix / Orientation Matrix

general, the iPod reproduces the "true" orientation of the device satisfactorily. A few of the elements in the histogram have low correlation coefficients. This may be explained by a low variance in the particular DCM element, which again causes a poor signal-to-noise ratio. The 8 recordings involved simple rotations around single axes, as well as composite rotations, with durations between 4 and 10 seconds.

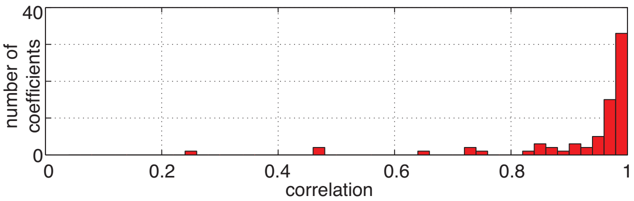


Figure 4: Histogram of correlation coefficients when correlating the orientation coordinates (DCM) of the iPod to Qualisys.

To analyse rotational drift, a gradient for each of the Euler angle coordinates was extracted by linear regression of the 10 minute recording of the iPod lying still in the motion capture space. A small amount of drift was observed in the orientation data. This, together with analysis of the rotational noise is shown in Table 2. The noise measurements are the RMS level of the Euler coordinates (in degrees), after subtracting the average drift, and centering around the mean value (removing offset). Note that compared to the Qualisys system, the iPod performs quite well when it comes to *roll* and *pitch*, with superior drift performance, and equal noise level, but the *yaw* measurements from the iPod are less accurate and less precise. The average yaw drift of a still recording is 70.6×10^{-5} deg/s which is equivalent to a drift of 2.5 deg/h. An additional effort to force the device to give inaccurate yaw data by shaking the device violently for 23 s, resulted in a yaw drift of 11.5 deg.

Table 2: Rotational drift and noise (Euler, degrees)

	Drift (10^{-5}deg/s)			Noise, RMS (= SD)		
	Roll	*Pitch*	*Yaw*	*Roll*	*Pitch*	*Yaw*
iPod	-0.61	1.05	70.6	0.028	0.018	0.153
Qualisys	-17.2	7.24	8.95	0.029	0.018	0.010

4.2.2 *Acceleration*

Acceleration data from the iPod is limited by the range of the accelerometer that provides the data. Apple has not officially released these specifications for the iPod, but it can quite easily be measured. Since the raw accelerometer data (which includes the gravity vector) and the user acceleration values are not identical, the range of acceleration values depends on the current orientation of the device. A recording of heavy shaking of the iPod provided maximum and minimum values of acceleration in the range -29 m/s^2 to $+29$ m/s^2, which is equivalent to ± 3 g.

Table 3 shows acceleration data statistics for the 10 minute recording of the iPod lying still. The table shows high standard deviations and max/min values for unfiltered Qualisys data. This is because even small noise in the position data will become large when the derivative is calculated [17]. However, a filtered version, using a simple 5 sample averaging filter on each derivative level significantly improves this. As shown, the iPod has a certain offset in this data, even though internal processing on the device is supposed to remove the gravity component in the acceleration data. The standard deviations from the iPod are slightly higher than the filtered Qualisys data.

Figure 5 shows that the acceleration data from the two systems match well (Qualisys is filtered as mentioned above). This will be discussed more in the next section.

Table 3: iPod acceleration noise, unit: 10^{-3} m/s^2

		mean	SD	min	max
iPod	*X*	5.3	18.5	-71.1	84.8
	Y	0.7	15.8	-67.9	61.8
	Z	160.7	22.6	33.9	303.8
Qualisys	*X*	0.005	261.5	-1613	1637
unfiltered	*Y*	0.001	272.4	-2431	2236
	Z	0.003	358.1	-2502	2745
Qualisys	*X*	0.000	10.4	-49.0	71.3
filtered	*Y*	0.000	12.3	-73.3	61.3
	Z	0.000	16.7	-77.6	87.3

4.2.3 *Position, Velocity, and Acceleration*

By integrating the acceleration data from the iPod, and differentiating the position data from Qualisys, we have estimated the accelerations, velocities and the positions measured by the two systems. Acceleration values from the iPod are given in local coordinates (cf. Section 3.2), while the second derivative of Qualisys position data provides acceleration in a global coordinate system. Hence, the iPod acceleration vector was transformed to a global coordinate system. This means that any orientational drift also influenced calculations of position.

Figure 5 shows an example of a short recording containing a simple vertical translation followed by a pitch rotation combined with vertical translation. The figure shows some drift in velocity, and a lot of drift in position. The figure also shows an attempt to correct for the positional drift through filtering, but long filters can induce unacceptable amounts of delay. In the figure, a 100 samples FIR filter is used, which corrects for some of the drift, but in most real-time settings a filter of this length would cause too much latency.

Figure 6 shows similar plots of the 10 minute still recording. There was a small offset of 0.16 m/s^2 in the iPod acceleration data, which was removed before estimating velocity and position. Even after removing the offset, the drift is significant. After one minute, the error of the position estimate is more than 1 m, and after 10 minutes, it is more than 60 m.

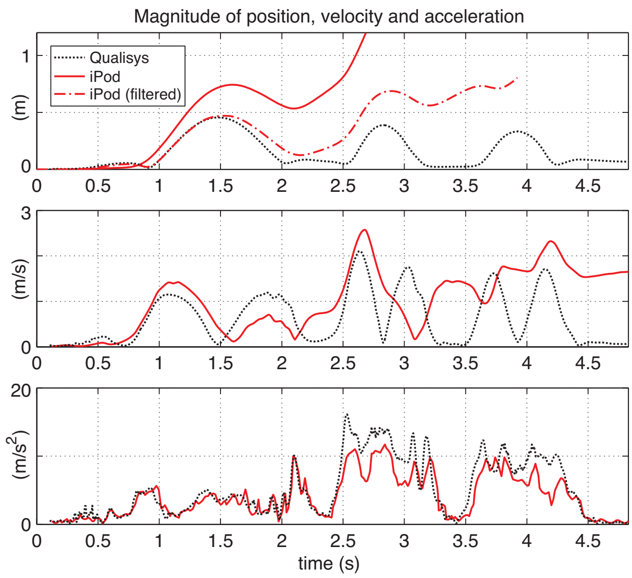


Figure 5: Plots of a short motion sequence, with magnitude of position, velocity and acceleration for iPod and Qualisys data. The filtered version of iPod position data has been time-shifted forward by 51 samples to compensate for filter latency.

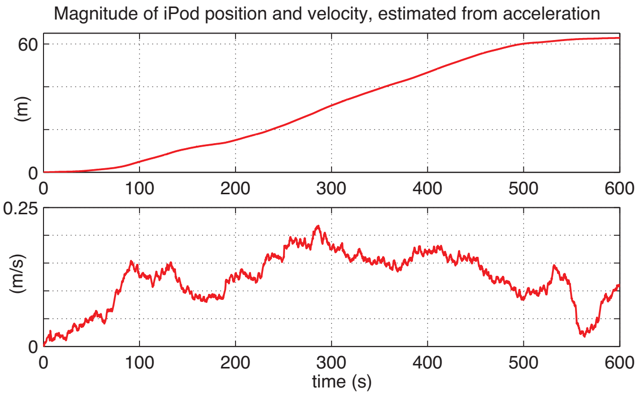


Figure 6: Plots of the magnitude of estimated position and velocity of the iPod lying still for 10 min.

5. DISCUSSION

We have presented how the motion data from an iPod compares to data from a high-end motion capture system. These results influence how we use the iPod motion data in music applications. Our analysis of lag in orientation data showed that there is an average of 43 ms between the time when an orientation is measured in the Qualisys system and when it is measured on the iPod. There may be several reasons for this; the different sampling rates of the two systems might have played some role, but we find it reasonable to assume that the processing done on the iPod to extract the orientation from the sensor raw data is the main cause. This means that it might be unsuitable to use orientation to control musical features that demand high temporal precision.

In addition to the lag, orientation data was evaluated in terms of accuracy and precision. For *roll* and *pitch* coordinates, the accuracy and precision are high, and sufficient for continuous control of sound. *Yaw*, on the other hand, does not show equally good results, and should be used with caution. The drift is still low enough to assume that it is suitable for measuring relative rotations over short time periods. In future work, it would be interesting to compare this with newer iPhone models which contain magnetometers.

The data jitter from the iPod is significantly higher than for Qualisys, despite the fact that the iPod sent less data at a lower sampling rate. This might be important to consider if the iPod is used for direct control of sound on a separate computer. The jitter could be compensated for by buffering, but this again would cause increased latency.

As expected, our attempt to estimate the position of the iPod from the acceleration data resulted in large errors, since the noise propagates a lot when the signal is integrated. Still, we notice that some positional features can be inferred from the iPod acceleration data. Especially for shorter segments, it is possible to tell when the iPod is moved in one plane, but the estimates are too imprecise to estimate when the device reaches back to the starting position. As seen in the lower plot in Figure 5, the acceleration data from the iPod is quite responsive, and is well suited for controlling musical parameters that require high temporal precision.

6. ACKNOWLEDGMENTS

This research has received funding from EU FP7 under grant agreement no. 257906 (EPiCS), and the Norwegian Research Council, project no. 183180 (SMA).

7. REFERENCES

[1] Apple Inc. iOs Developer Library, CMDeviceMotion. `http://developer.apple.com/library/IOs/#documentation/CoreMotion/Reference/CMDeviceMotion_Class/Reference/Reference.html`.

[2] P. Brinkmann, P. Kirn, R. Lawler, C. McCormick, M. Roth, and H.-C. Steiner. Embedding pure data with libpd. In *Proceedings of the Pure Data Convention*, Weimar, Germany, 2011.

[3] N. J. Bryan, J. Herrera, J. Oh, and G. Wang. MoMu: A mobile music toolkit. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 174–177, Sydney, 2010.

[4] N. J. Bryan and G. Wang. Two turntables and a mobile phone. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 179–184, Oslo, 2011.

[5] G. Essl. UrMus – An environment for mobile instrument design and performance. In *Proc. of the International Computer Music Conference*, pages 270–277, New York, 2010.

[6] G. Essl and M. Rohs. Mobile STK for Symbian OS. In *Proc. of the International Computer Music Conference*, pages 278–281, New Orleans, 2006.

[7] G. Essl and M. Rohs. ShaMus – A sensor-based integrated mobile phone instrument. In *Proc. of International Computer Music Conference*, pages 27–31, Copenhagen, 2007.

[8] G. Essl and M. Rohs. Interactivity for mobile music-making. *Org. Sound*, 14:197–207, 2009.

[9] G. Essl, G. Wang, and M. Rohs. Developments and challenges turning mobile phones into generic music performance platforms. In *Proc. of Mobile Music Workshop*, pages 11–14, Vienna, 2008.

[10] G. Geiger. PDa: Real time signal processing and sound generation on handheld devices. In *Proc. of Int. Computer Music Conference*, Singapore, 2003.

[11] A. R. Jensenius. Some challenges related to music and movement in mobile music technology. In *Proc. of Mobile Music Workshop*, pages 19–22, Vienna, 2008.

[12] N. Kruge and G. Wang. MadPad: A crowdsourcing system for audiovisual sampling. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 185–190, Oslo, 2011.

[13] K. Nymoen and A. R. Jensenius. A toolbox for storing and streaming music-related data. In *Proc. of Int. Sound and Music Computing Conference*, pages 427–430, Padova, 2011.

[14] J. Oh, J. Herrera, N. J. Bryan, L. Dahl, and G. Wang. Evolving the mobile phone orchestra. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 82–87, Sydney, 2010.

[15] G. Schiemer and M. Havryliv. Pocket Gamelan: a PureData interface for mobile phones. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 156–159, Vancouver, 2005.

[16] E. Schubert. Correlation analysis of continuous emotional response to music. *Musicae Scientiae*, Special issue 2001–2002:213–236, 2002.

[17] S. A. Skogstad, K. Nymoen, and M. E. Høvin. Comparing inertial and optical mocap technologies for synthesis control. In *Proc. of Int. Sound and Music Computing Conference*, pages 421–426, Padova, 2011.

[18] A. Tanaka. Mobile music making. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 154–156, Singapore, 2004.

[19] G. Wang. Designing Smule's Ocarina: The iPhone's magic flute. In *Proc. of Int. Conf. New Interfaces for Musical Expression*, pages 303–307, Pittsburgh, 2009.