PUBLICATION P3

Ilmonen, Tommi and Jalkanen, Janne. Accelerometer-Based Motion Tracking for Orchestra Conductor Following. Proceedings of the 6th Eurographics Workshop on Virtual Environments, Amsterdam, Netherlands 2000.

Accelerometer-Based Motion Tracking for Orchestra Conductor Following

Tommi Ilmonen and Janne Jalkanen

Helsinki University of Technology
Telecommunications Software and Multimedia Laboratory
P.O.Box 5400
FIN-02015 HUT
Tommi.Ilmonen@tcm.hut.fi
Janne.Jalkanen@tcm.hut.fi

Abstract. In this paper, we discuss the applicability of accelerometers to measuring movement for conductor following. In our application a baton is used to conduct computer animated musicians. The user acts as an orchestra conductor. The baton motion is analyzed for gestures that imply how music and animation should be controlled. The baton motion is tracked with accelerometers. The accelerometers feature inevitable problems for position tracking. Position cannot be measured directly – it needs to be integrated twice from the acceleration. The measurement inaccuracy causes drift over time when velocity and position are calculated via integration. We used two sensors to track motion with six degrees-of-freedom. The problems can be largely overcome by using application-specific signal processing. The drift caused by inaccurate integration is countered by combining leaky integrators and high-pass filters. Rotation is detected by monitoring the direction of gravity.

1 Introduction

Our application is a conductor follower system [1]. Using this application a user can conduct a virtual band with a baton. The system has been developed using a magnetic tracker as the motion measurement tool. The high price of the hardware became a problem when a low-cost version of the system was needed, so we developed an alternative motion tracking mechanism using accelerometers. The system performs partial motion tracking, as it only detects relative motion in a vertical plane. Similar inertial motion sensing mechanisms have been studied by Verplaetse [2]. Marrin has also used accelerometers with a baton [3]. Those systems concentrated more on motion sensing than motion tracking.

Inertial motion tracking has strict limitations [4]. If one can only measure acceleration the velocity and position values are obtained via integration. The integration makes the accelerometer approach sensitive to any errors in measurement or initial position estimation. The errors accumulate over time and cause drift. Due to this drift will eventually be a problem – regardless of the accuracy of the measurement.

In our application drift is very harmful, but other forms of error are tolerable. A digital signal processing (DSP) net (Fig. 2) can transform the artifacts to less troublesome form. Knowledge of the application and situation is used to construct the net. Our application is sensitive to coordinate orientation errors and long-term position errors. These problems are targeted with the DSP.

2 The Application

The motion tracking is one component of the DIVA (Digital Interactive Virtual Acoustics) virtual band [5]. The system is an orchestra simulator – the user conducts the band by waving a baton like a real orchestra conductor. The band reacts to the user by playing the pieces as conducted [6].

Motion tracking is needed since the system must know how the conductor is moving. The motion data is used by our conductor following software for gesture analysis. The movements are analyzed and extrapolated by artificial neural networks and heuristic rules in order to time the notes correctly. The detected movements are a subset of orchestra conducting movements (for example [7]). The system can be tuned for each individual conductor, piece and motion tracking method via configuration files.

A stripped low-cost version was needed because a local science center "Heureka" wanted a show stand featuring the band. On the stand a single computer runs band animation and conductor following, while visitors can conduct the band.

3 System Requirements

The public exhibition of a virtual orchestra poses several requirements for the software and the hardware. Since the stand is not guarded, and a lot of the visitors are school children, the physical strain on the system is considerable, and the baton must endure severe mis-use. Hardware failures will be common in such surroundings, and thus every part of the system must be easily and cheaply replaceable.

We chose a standard PC to be our hardware platform, enhanced with an off-the-shelf analog to digital (AD) card (ACL-8112), a common 3D card (based on the 3dfx Voodoo chip set), and a sound card for MIDI output. The computer is hidden to prevent tampering. Most of the user interface is also hidden from the user. The only input device beside the baton is a series of heavy-duty buttons used to select the piece of music. The music is played by a synthesizer module and controlled through MIDI events.

The software must be robust, and very easy to use. We chose Linux to be the operating system, as it fit well with our development environment (SGI, C++, Tcl/Tk, OpenGL), and the kernel (2.0 at that time) is very stable. Porting our existing software proved to be easy, and the customizability of Linux allowed us to tailor the system for the particular needs of the science center.

4 The Sensors

Each accelerometer measures lateral acceleration (VTI SCA600 accelerometer chip [8]). Multiple sensors are bundled to measure multidimensional movement. The sensors are powered externally, and the signal cable doubles as power cable.

The sensors have a single analog output that gives the acceleration of the system in form of voltage variations. The AD-card in the computer is used to measure the voltages. The accelerometers have limited measurement range, and available ranges were 3g and 50g. Outside the valid range the sensor output clips to the extreme value. Although the 3g range is not enough to cover strongest movements we decided to use it because of the better signal-to-noise ratio. The sensors can withstand a 20 000 g acceleration without breaking (ie. they survive hits well).

All sensors are slightly different in their output during zero acceleration. This is a problem that cannot be compensated in real-time. The software is calibrated on-site to match the zero-acceleration direct current (DC) level of the accelerometers in use.

5 Preparing The Baton

The baton is a hollow plastic tube, 15 mm in diameter, with a cable coming out from one end. The accelerometers are inside the tube, mounted on a piece of wood, since no circuit boards fit inside the tube. In order to lessen the shock from impacts the accelerometers are also hermetically sealed with a soft rubber tube before they were put inside the plastic tube (Fig. 1). The baton is approximately 25 cm in length, weighing considerably more than a standard conductor's baton, but being still light enough for small children to handle. However, the extra mass adds stability, as the small jitter from the users hand is somewhat compensated by the inertia of the baton. The wiring poses more of a problem for the conductors than the baton weight.

Two accelerometers are mounted orthogonally to measure vertical and horizontal acceleration. A third accelerometer was included to measure depth-directional movement, but it turned out that sufficient information could be obtained with only the vertical and horizontal sensors. This can be attributed to the fact that the conductors movements mostly consist of vertical and horizontal movements. Since the baton movements make a roughly spherical surface, the third sensor could be used to measure centripetal acceleration.

The sensors are placed approximately 5 cm from the near end of the tube. While placing the sensors near the fast-moving tip of the baton would give more accurate results, it would also subject the sensors to much higher accelerations. Since the limit of the used accelerometers was only 3g, we had to place them as near the users hand as possible. However, the sensors could not be placed at the very bottom, since some users tend to conduct by moving their wrist only. Sufficient movement for calculation could be received at the approximate position of the users thumb.

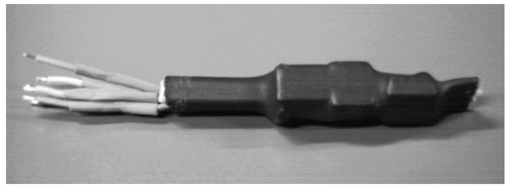

Fig. 1. A prototype with three sensors mounted together. Only two sensors are used in the current system.

Wiring needs to be light for the first 1.5 meters, and thus is not shielded. The rest of the cable is shielded. We have not had any problems with outside interference.

6 Heuristics And Related Dsp

The accelerometer is a non-ideal motion tracking tool: Measurement errors or offsets cause drift to position estimation. With no rotation detection it is not possible to detect which way the baton is held. If the baton orientation is not known the constant offset caused gravity cannot be countered.

The application does pose some helpful restrictions, since we are measuring only a limited set of movements within a tightly confined space. Accurate position information is not needed, since speed, relative position, and acceleration are more important. Even big errors in measurement are allowed as long as the form of the motion is presented correctly. Figure 2 shows the DSP net that is used to counter these problems.

6.1 Drift

The inherent problem of accelerometer-based motion tracking is the drift. Double integration of the measurements guarantees any noise in the input will eventually make position estimation inaccurate. This applies to both position and orientation measurements.

The first graph in figure 3 illustrates the problem of integrating a variable with constant offset (or with incorrect initial value guess) – eventually the integrated value will reach infinity. The drift caused by integrating noisy measurements can be removed by assuming long-term zero average. If the value obtained via integration is constantly well below or above zero an offset value is added. Since the motion is limited we can make this assumption safely.

The most simple method to determine the constant offset is to calculate the mean value of the integrated signal during some period of time. By subtracting the mean from the integrated signal one removes the offset. This procedure has some problems and it needs to be reformulated to be useful. The first problem is the integration. One would rather not calculate the integral at all since it

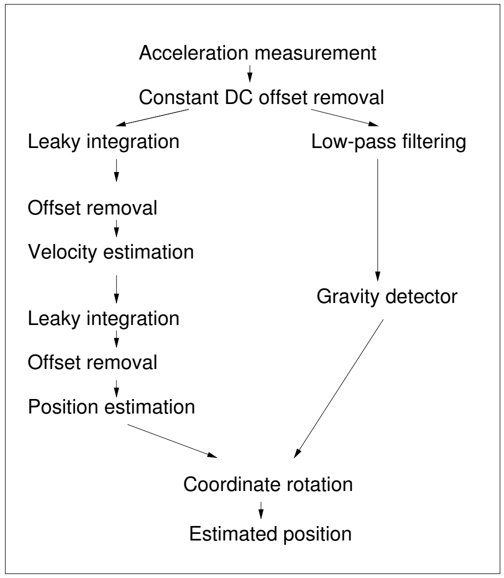

Fig. 2. Signal flow.

will eventually reach infinity (numeric problems). The second problem is that calculating the mean of the signal for significant number of samples consumes a CPU time (computational cost).

A signal processing approach can be used for the integration and offset removal. With leaky integration (low-pass filter) the effect of constant offset can be partially compensated. A high-pass filter is needed to keep the integrated signal close to zero. Figure 3 shows how the results change when the integrator is made leaky (integration and offset removal) and a high-pass filter (constant offset removal) is added to the integrator output. A leaky integrator is effectively an integrator with offset removal. Since the mean of the integrated signal increases linearly the second offset removal is needed.

In our implementation both the leaky integrator and the high-pass filter are simple first order infinite impulse response (IIR) filters (Fig. 4). A single low-pass filter (leaky integrator) performs both integration and offset removal. The low-pass filter has gain coefficient A_1 close to unity, B_0 equals one and B_1 is zero. The high-pass filter has A_1 and B_0 close to unity and $B_1 = -B_0$. The graphs in figure 3 illustrate results obtained with these filters.

Cascading two first-order filters produces effectively one second order filter. We have used two separate filters because it is easier to monitor the intermediate results that way. These intermediate results are useful when the parameters of the net are optimized.

The filter design procedure differs from the normal filter design methods ([9], chapter 8). While usually a filter is designed around the concept of frequency we

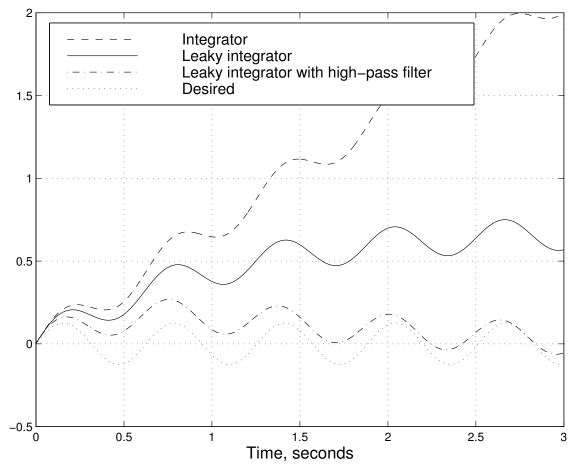

Fig. 3. Using DSP to remove integration artifacts (numeric simulation). The resulting signal is slightly distorted.

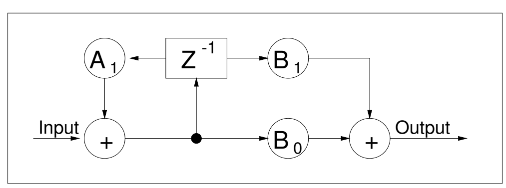

Fig. 4. First order all-pole filter. The block Z^{-1} is single-sample delay.

design them to perform integration and offset removal. Low phase distortion and faithful reproduction of the original waveform are crucially important. It could be possible to construct explicit methods to calculate filter parameters (resembling control theory), but we simply experimented with numeric values until we found a good combination. With such a simple problem these experiments do not take much time. The frequency responses of these filters are illustrated in figure 5. In both cases the cut-off frequencies are between 0.1 and 0.3 Hertz.

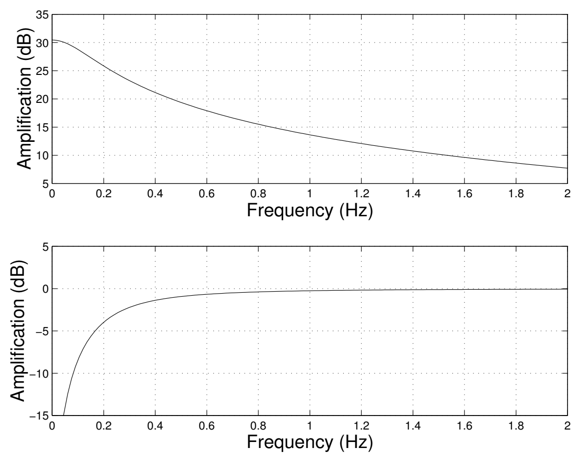

Fig. 5. Low frequency responses of the integrator and high-pass filters.

6.2 Orientation

The procedure outlined above yields position estimations towards two axis. To understand the movement we need to know what are the directions of the axis ie. the rotation of the baton. We have used gravity as the key to orientation detection. Gravity manifests itself as persistent acceleration towards a single direction. A low-pass filter is used to calculate the average vector of the two acceleration measures. The filter is like the low-pass filter that performs leaky integration, only with slightly different parameters. The resulting vector estimates the direction of the gravity and that information yields the rotation of the baton.

New orientation estimate is calculated at each sample, and the coordinates obtained via double integration are then rotated according to this estimate. If the rotation estimate was update more seldom transients would result as the angle is updated. The low-pass filtered acceleration has much amplitude than the original signal (Fig. 6). Since the orientation value changes smoothly and slowly, changing rotation does not cause transients to the final coordinate output.

While this works most of the time there are conditions when the system falls short. If the user twists his/her wrist during operation the filtering will produce bad rotation estimates for a few seconds. After those few seconds the orientation estimate is correct again. Luckily people seldom rotate the baton aggressively.

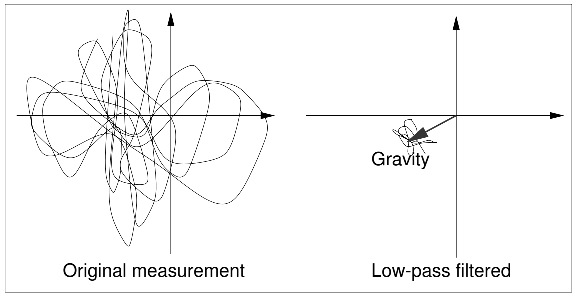

Fig. 6. Gravity detection.

6.3 Aliasing and Clipping

Values from the accelerometers need to be sampled at fixed sampling rate. As a rule of a thumb, the higher the rate the better the results. Short acceleration bursts may go unnoticed by the system if the sampling rate is too low (Fig. 7). In those cases the signal will be aliased, which causes distortion to the final position estimation. The mass of the baton helps us in this case, as it increases the inertia and smoothes the natural tremor by the user's hands. With the user holding only the accelerometers the measurement results were very noisy, but when the whole of the baton was used measurements became much more clean. In our application 33 Hz sampling rate is adequate. This gives Nyquist frequency of 16.5 Hz (higher frequencies will be aliased). Our tests imply this is a sufficiently high Nyquist frequency ([6], page 25) and it also exceeds the 12 Hz limit given by Verplaetse [2].

There are special circumstances when the signal will alias. The most common of these is the case when the baton hits a rigid object - the stand for example. In these cases the baton will experience a short, very strong acceleration peak. Handling these well would require sampling rates in excess of one kHz. The accelerometer should also be able to measure acceleration in excess of 100 g. In practice these situations are rare enough not to cause any real problems, and thus we do not take them into account.

A more practical problem stems from the 3g range limit. Human arm can easily produce up to 10g accelerations. When this happens the output signal from the sensor clips, which causes a major measurement error. If the integrators did not leak this would be a serious problem. The leakiness of the integrators and the high-pass filters remove most of the artifacts. No special processing is needed for such cases. The system can be conducted well even when the sensors' output clips.

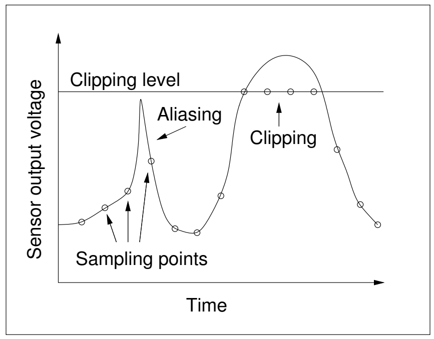

Fig. 7. Signal aliasing and clipping - significant information is lost due to too low sampling rate and signal clipping.

The sampling must be done in steady rate. This can be a problem in time-shared multitasking system. We use real-time priorities to guarantee the operating system scheduler allocates CPU-time when needed. The software reads the current values of AD-cards inputs 33 times per second. This means the software must access the AD-card every 30 milliseconds. The interval is chosen to be an exact multiple of the 10 millisecond scheduling interval of the Linux/x86 platform, minimizing scheduling jitter.

7 Conclusions

For our application the results are good – a durable motion tracking system can be built at low cost. It is not comparable to a more expensive magnetic, optical or ultrasound trackers. The lack of absolute position or orientation information can be a difficult problem in other applications. The final position estimate is distorted, but the gesture analysis application is robust enough to handle the distortion. Thorough analysis of the performance of the motion tracking system (accuracy, limits) has not been done. In our case the performance metrics are not so obvious, simple location errors do not tell how big problems the application will have.

Some problems are inevitable. The leaky integrators and high-pass filters also affect the motion shape. The slower the movement the more notable this effect is. This is a problem when very slow tempos (below 50 beats per minute) are conducted.

The most difficult problem we have experienced is the durability of the wiring, and the soldering on the batons. Continuous use and shocks cause the wiring to break off every few weeks, and the wiring must be replaced. Also, sometimes the accelerometers must be re-soldered when contacts become loose.

As far as software and implementation go the system has good performance. With all the filtering and AD-card access the application uses less than 2% of the systems CPU-time. Neither does it harm multitasking.

There is room for improvements, though: Adaptive filter parameter adjustment might improve the performance. A thorough analysis of measurement errors would help determine the optimal parameters for the DSP network. Adding rotation sensors would improve accuracy, but also add price and failure rate.

8 Acknowledgments

This work was funded by Finnish Technology Development Centre Tekes, the Finnish Science Centre Heureka, Veikkaus, Okmetic and VTI Hamlin.

References

1. Ilmonen, T. and Takala, T. . Conductor following with artificial neural networks. In Proceedings of the International Computer Music Conference, pages 367–370, 22-28 October 1999. URL: http://www.tcm.hut.fi/Research/DIVA/publications/1999/ilmonen_icmc99.ps.gz.
2. Verplaetse, C. . Inertial proprioceptive devices: Self-motion-sensing toys and tools. IBM Systems Journal, 35(3&4):639–650, 1996.
3. Marrin, Teresa . Toward an understanding of musical gesture: Mapping expressive intention with the digital baton. Master's thesis, MIT, 1996.
4. Britting, K. R. . Inertial Navigation Systems Analysis. Wiley & Sons, Inc., New York, 1971.
5. Hiipakka, J. , Hänninen, R. , Ilmonen, T. , Napari, H. , Lokki, T. , Savioja, L. , Huopaniemi, H. , Karjalainen, M. , Tolonen, T. , Välimäki, V. , Välimäki, S. , and Takala, T. . Virtual orchestra performance. In Visual Proceedings of SIGGRAPH'97, page 81, Los Angeles, 1997. ACM SIGGRAPH.
6. Ilmonen, T. . Tracking conductor of an orchestra using artificial neural networks. Master's thesis, Helsinki University of Technology, Telecommunications Software and Multimedia Laboratory, April 1999.
7. McElheran, B. . Conducting Technique for Beginners and Professionals. Oxford University Press, Oxford/New York, 1989.
8. Hamlin, VTI . VTI Hamlin Acceleration Sensors. Document named sca600.html available via internet from the server "www.vti.fi". Visited on 15.2.2000. URL: http://www.vti.fi/Products/Sensors/SCA600/sca600.html.
9. Proakis, J. G. and Manolakis, D. G. . Digital Signal Processing. Macmillan Publishing Company, New York, 1992.