Chapter 12
Remote Hand Motion Detection and Monitoring with Noise Reduction

Jing Pang

Abstract The digital triaxial accelerometer has become more and more popular in the research and industrial world due to its small size, low power consumption, low cost and its sensing capability in the x, y and z-axis directions. This paper presents several noise reduction schemes for hand motion detection with the triaxial digital accelerometer ADXL345 by using the RCM3365 board as the web server for control and also for data monitoring.

Keywords Digital accelerometer · Hand motion · Kalman filter · Median filter · Moving average filter · Noise reduction · Output data rate · Remote data monitoring · Web server

12.1 Introduction

The triaxial accelerometer based hand motion detection has a wide range of potential applications, including handsets, mobile device, gaming, personal navigation devices, learning, sports medicine, health-care and other gesture-based customer devices.

The small size integrated microelectromechanical system (iMEMS) accelerometer such as ADXL345 [1] has an integrated on-chip analog-to-digital converter (ADC) and it provides digital outputs with SPI and I^2C digital interface signals. Such accelerometers can be connected with the microcontroller through the SPI or I^2C

J. Pang (✉)
Department of Electrical and Electronic Engineering, Computer Engineering Program,
California State University, 6000 J Street, Sacramento, CA 95819, USA
e-mail: pangj@gaia.ecs.csus.edu

H. K. Kim et al. (eds.), *IAENG Transactions on Engineering Technologies*,
Lecture Notes in Electrical Engineering 170, DOI: 10.1007/978-94-007-4786-9_12,

interface, and can be used for acquiring x-axis, y-axis and z-axis direction sensing data. Usually, one limitation of microcontrollers is that they have limited memory for data storage. As a result, this design work used the RCM 3365 module as a web server to control the ADXL345 device and monitor hand motion. At the same time, the remote PC could obtain the acquired accelerometer data through Ethernet and then store them into the Excel database in real time for remote monitoring.

One 8-bit Rabbit 3000 microprocessor is mounted on the RCM3365 module [2] and it runs with a maximum clock frequency of 44.2 MHz. RCM3365 has 512 K Flash, 512 K program SRAM, and 512 K data SRAM. It works with a DC power supply that ranges from 3.15 to 3.45 V. With a 3.3 V power supply and a 44.2 MHz clock, the current consumption is about 250 mA. RCM3365 supports Dynamic C with the royalty-free TCP/IP stack to enable rapid and secure web interface development.

A C# program running on a remote PC was developed to acquire real-time accelerometer data on the internet and also to store data in Microsoft Excel sheets on the remote PC. Noise reduction is important for getting reliable and high resolution triaxial accelerometer measurements. This is especially true for some sensitive applications that require a high level of precision. The major noise sources for the surface technology silicon digital accelerometers are from Brownian mechanical noise, electronic thermal noise, and internal analog to digital conversion quantization noise.

The noise reduction methods considered in this work include techniques to be implemented on the hardware in real time and also techniques for post processing using software.

The rest of the paper consists of multiple sections. Section 12.2 describes the hardware system. Section 12.3 explores multiple real-time noise reduction techniques, including using different output data rates, applying a median filter [3] and applying a moving average filter [4]. Section 12.4 discusses the Kalman filter [5] noise reduction technique for post processing. Finally, Sect. 12.5 gives conclusion.

12.2 Hardware System

Several hardware devices are used in the remote hand motion detection system shown in Fig. 12.1. They are the accelerometer ADXL345, the RCM3365 module, the NETGEAR wireless router, and the PC. The ADXL345 is used to detect hand motion. The RCM3365 communicates with the ADXL345 through an I^2C interface to acquire hand motion data. The NETGEAR wireless router connects the RCM3365 with the internet so that the server data stored inside the RCM3365 module can be posted on the website in real time. Moreover, the remote PC gets access to the website accelerometer data and stores them into the Excel database in real time.

In Fig. 12.1, the ADXL345 chip works with a 3.3 V power supply. The clock signal SCL and the data signal SDA are connected with 2 K ohm pull-up resistors

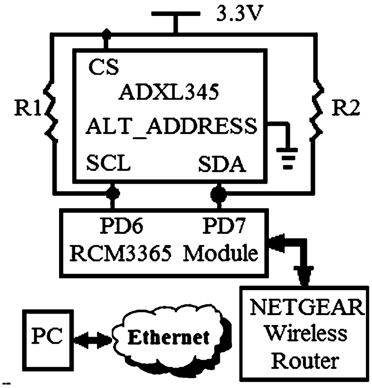

Fig. 12.1 Remote hand motion detection and monitoring hardware system

R1 and R2. They are controlled by the RCM3365 module through the I^2C interface. The I^2C mode is enabled when the CS pin is tied high to 3.3 V. The ALT_ADDRESS pin is the alternate I^2C address selection signal. The alternate I^2C address of 0x53 is chosen when the ALT_ADDRESS pin is grounded and this translates to 0xA6 for writing and 0xA7 for reading. By default, the Rabbit 3000 microprocessor on the RCM3365 module uses the PD6 and PD7 pins for I^2C communication. They are connected with the SCL and SDA pins of ADXL345 respectively.

The ADXL345 can be attached to the top of the user's hand. Six different hand positions have been tested successfully based on my previously conducted work: straight up, left-tilted facing up, left-tilted facing down, upside down, right-tilted facing down, and right tilted facing up [6]. In my previous work, the proposed algorithm covers a relatively wide angle range for each hand motion position and it is not sensitive to small sensor errors. The wireless router NETGEAR is used to interface the RCM3365 with the Internet. The RCM3365 serves as the web server to control and monitor hand motion data collected by the accelerometer. The remote PC can access the hand motion accelerometer data through the Ethernet using the C# program, and then store them into the Excel database in real time.

In order to detect a more accurate accelerometer angle for hand motion, noise reduction techniques need to be studied. At the same time, the noise reduction techniques need to consider the tradeoff between the algorithm complexity and the real-time angle detection.

12.3 Real Time Noise Reduction Schemes

Three schemes were implemented in this design work to study the effect of real-time noise reduction including: changing the RCM3365 output data rates, applying a moving average filter and applying a median filter.

Table 12.1 The BW_RATE 0x2C register map [2]

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	LOW_POWER	Rate			

12.3.1 BW_RATE Register and Rate Code

The BW_RATE 0x2C register map of the ADXL345 chip is shown in Table 12.1. The rate bits set the output data rate and the LOW_POWER bit sets the power mode.

In this design work, the LOW_POWER bit in the BW_RATE register is set to 0 to allow the ADXL to work in the normal mode to reduce noise. According to the ADXL345 datasheet, when the LOW_POWER bit is set to 1, the ADXL operates with reduced power consumption. However, the noise is somewhat higher.

In order to check the relationship between the output data rate and the noise level, several output data rates were tested: 400, 200 and 100 Hz. The rate codes are set according to the Table 12.2.

12.3.2 Performance Comparison

Figures 12.2, 12.3 and 12.4 illustrate the hand motion detection in the x-axis, y-axis, and z-axis directions with the RCM3365 output data rate set to 400 Hz.

For Fig. 12.2 the mean and standard deviation of accelerometer x-axis angle are 1.123325 and 0.244906, respectively.

For Fig. 12.3 the mean and standard deviation of accelerometer y-axis angle are 5.260033 and 0.372795, respectively.

For Fig. 12.4 the mean and standard deviation of accelerometer z-axis angle are 5.385482 and 0.363293, respectively.

In order to reduce noise, both the moving average filter and the median filter applied a window size of four to the accelerometer data samples. According to the results shown in Table 12.3, both methods get similar performance results and they achieve much smaller standard deviation values compared with the original x-axis, y-axis, and z-axis data.

Since the ADXL345 hardware itself presumably supports mechanisms internally to improve the accuracy of the measurements when using a lower sampling rate, different output data rates were tested, along with a moving average filter and a median filter for a window size equal to 4. The results are shown in Figs. 12.5, 12.6, and 12.7. These figures show that reducing the output rate also reduces the standard deviation of the noise in the measurements. With the window size equal to four, either the moving average filter or the median filter can achieve the similar performance to the method of changing the output data rate to four times smaller.

Table 12.2 Output data rate and rate code [2]

Output data rate (Hz)	Rate code
400	1100
200	1011
100	1010

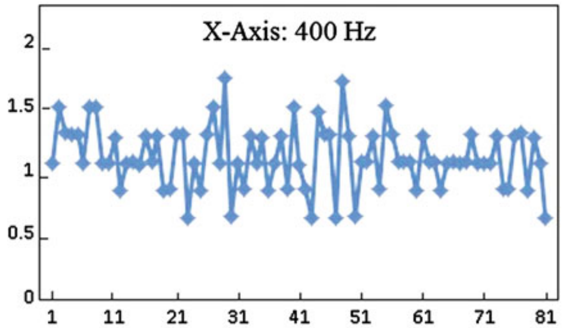

Fig. 12.2 Hand motion detection in the x-axis with the output data rate of 400 Hz

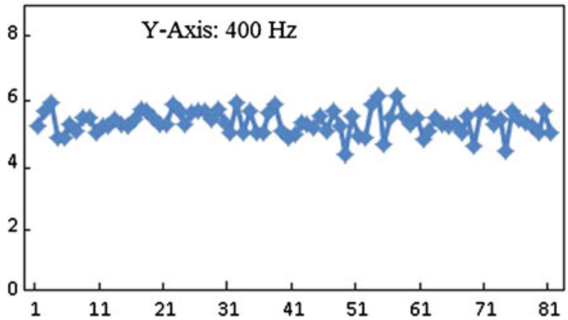

Fig. 12.3 Hand motion detection in the y-axis with the output data rate of 400 Hz

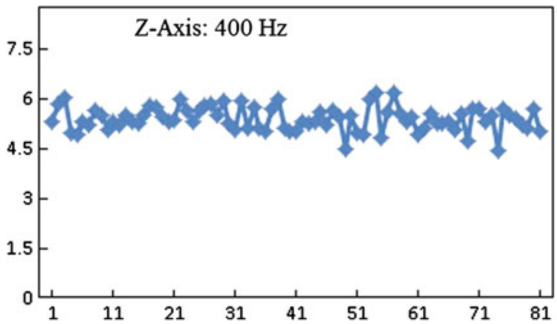

Fig. 12.4 Hand motion detection in the z-axis with the output data rate of 400 Hz

Figure 12.5 shows that if the moving average filter or the median filter is applied to the accelerometer data, the additional ADXL345 output data rate reduction will not lower the measurement noise standard deviation a lot.

Table 12.3 Mean and standard deviation for accelerometer data with output data rate 400 Hz

	Original data	Moving average	Median
X-axis mean	1.123325	1.127214	1.130681
X-axis standard deviation	0.244906	0.09544	0.111451
Y-axis mean	5.260033	5.259894	5.267801
Y-axis standard deviation	0.372795	0.167936	0.187202
Z-axis Mean	5.385482	5.386154	5.392974
Z-axis standard deviation	0.363293	0.166385	0.186787

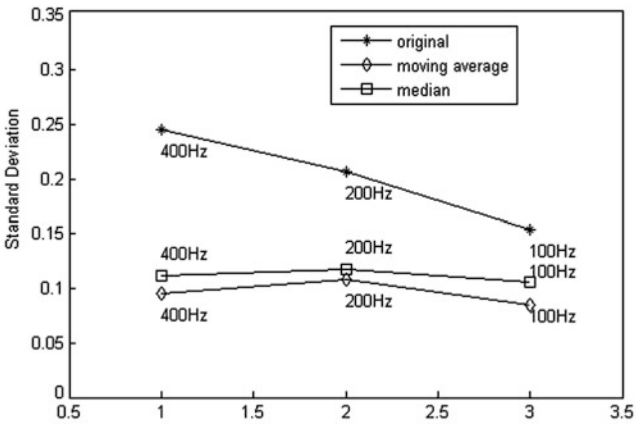

Fig. 12.5 Standard deviation for hand motion detection schemes in the x-axis direction

Figure 12.6 looks similar to Fig. 12.7 because y-axis and z-axis direction angles are similar in the testing case. Since the hand is approximately still, the mean and standard deviation of both angles show similar values.

Both the moving average and median filters smooth the input data high frequency noise. The median filter is especially good for smoothing the impulse noise. For a small window size of 4, only a small amount of storage is required and it allows for the fast computation speed for the moving average and median filter implementations. However, if the window size becomes bigger, both the moving average and median filters will create data lags.

12.4 Kalman Filter

In comparison to the moving average filter and the median filter methods which require past measurements, the Kalman filter uses only the present measurements and the previously calculated state [7].

The measurements from accelerometers are perturbed by noise. In this work, all process and measurement errors are assumed to have Gaussian distributions. The

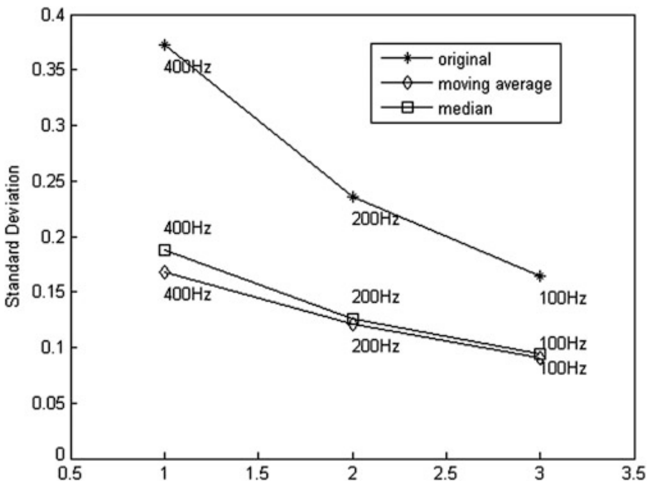

Fig. 12.6 Standard deviation for hand motion detection schemes in the y-axis direction

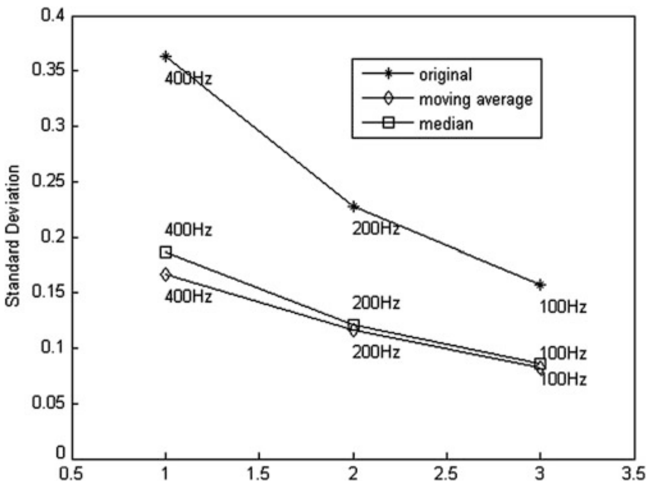

Fig. 12.7 Standard deviation for hand motion detection schemes in the z-axis direction

accelerometer measurement is modeled as a linear dynamical system according to the Kalman filter framework in the following equations. X(t) is the vector that includes the accelerometer x-axis, y-axis, z-axis angle measurements as well as the angular speed signals: mx(t), my(t), mz(t), vx(t), vy(t), and vz(t). X(t) and X(t − 1) are the internal states of the process at time t, and time t − 1 respectively. X(t) is represented as:

$$X(t) = F^{*}X(t-1) + W(t) \quad (12.1)$$

$$F = \begin{bmatrix} 1 & 0 & 0 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 & dt & 0 \\ 0 & 0 & 1 & 0 & 0 & dt \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{12.2}$$

$$X(t)=[mx(t)my(t)\ mz(t)\ vx(t)\ vy(t)\ vz(t)]^T \tag{12.3}$$

$$dt = 1 \tag{12.4}$$

Where W(t) is the process noise which has zero mean and covariance Q, All speed signals are normalized based on the time interval of dt equal to 1.

Moreover, Z(t) is the vector representing the noisy accelerometer x-axis, y-axis and z-axis angle measurements according to the following equations.

$$Z(t) = H * X(t) + V(t) \tag{12.5}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 & dt & 0 \\ 0 & 0 & 1 & 0 & 0 & dt \end{bmatrix}; \tag{12.6}$$

Where V(t) is the measurement noise which has zero mean and covariance M.

To solve the above equations, the Kalman filter provides computational means to obtain the state of the process using the prediction Eqs. (12.7) and (12.8), the Kalman gain Eq. (12.9) and the update Eqs. (12.10), (12.11), and (12.12) recursively. The P matrix is the error covariance matrix and the final filtered output matrix is OUTX. The Kalman gain is optimized to minimize the error covariance matrix. One advantage of the Kalman filter is that the new state estimate only requires the estimated state from the previous time step and the current measurement.

$$X_{prediction} = F * X_{estimation} \tag{12.7}$$

$$P_{prediction} = F * P_{estimation} * F^T + Q \tag{12.8}$$

$$K_{gain}=\left(H * P_{prediction}{}^T\right) * \left(H * P_{prediction}{}^T * H^T + R\right)^{-1} \tag{12.9}$$

$$X_{estimation} = X_{prediction} + K_{gain} * \left(Z - H * X_{prediction}\right) \tag{12.10}$$

$$P_{estimation} = P_{prediction} - K_{gain} * H * P_{prediction} \tag{12.11}$$

$$OUTX = H * X_{estimation} \tag{12.12}$$

In Figs. 12.8, 12.9 and 12.10, the Kalman filter is applied to the same data used in Figs. 12.2, 12.3 and 12.4 with the output data rate of 400 Hz.

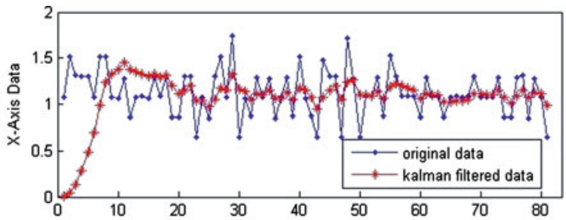

Fig. 12.8 Kalman filtered hand motion detection in x-axis direction

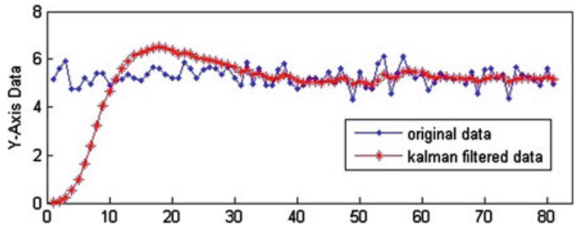

Fig. 12.9 Kalman filtered hand motion detection in y-axis direction

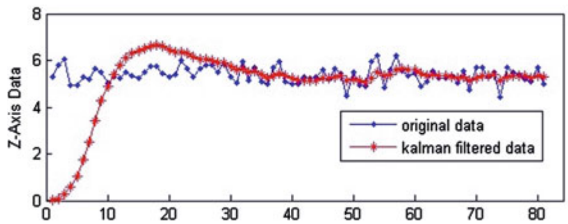

Fig. 12.10 Kalman filtered hand motion detection in z-axis direction

In Fig. 12.8, the mean is 1.1074 and the standard deviation is 0.0688 for the x-axis accelerometer data with the output data rate of 400 Hz.

In Fig. 12.9, the mean is 5.1836 and the standard deviation is 0.1374 for the y-axis accelerometer data with the output data rate of 400 Hz.

In Fig. 12.10, the mean is 5.3082 and the standard deviation is 0.1362 for the z-axis accelerometer data with the output data rate of 400 Hz.

Compared with the other schemes used in Table 12.3, the Kalman filter method gets the best standard deviation for the filtered result based only on the present measurements and the previously calculated state.

12.5 Conclusion and Future Work

Different noise reduction schemes are studied in this paper for hand motion detection using the accelerometer ADXL3365 and the rabbit RCM3365 module. In general, reducing the output data rate lowers the noise level. For real-time operation, both the moving average filter and the median filter techniques require the small window size in order not to get data lag. For the small window size such as four samples, the moving average filter and the median filter techniques have similar performance when they are applied to the accelerometer data. The Kalman filter scheme has the best standard deviation compared with the other methods. Furthermore, it is based only on the present measurements and the previously calculated state, which will allow for the real-time implementation of the noise reduction.

In this work, the different noise reduction schemes for hand motion detection are studied. However, the advantage of the Kalman filter scheme is only studied for the data post processing using software. In the future, further study will be explored to implement the Kalman filter scheme on hardware in real-time to allow noise reduction and tracking of hand motion for more complicated gesture recognition.

References

1. Analog Devices Inc., 3-Axis, ±2 g/±4 g/±8 g/±16 g Digital Accelerometer ADXL345, Product Datasheet, Norwood, MA (2011)
2. Digi International Inc., RabbitCore RCM3365/RCM3375 C-Programmable Core Module with NAND Flash Mass Storage and Ethernet, Product Manual, Minnetonka, MN (2008)
3. Arce GR (2005) Nonlinear signal processing: a statistical approach. Wiley, New Jersey
4. Demosthenous P, Nicolaou N, Georgiou J (2010) A hardware-efficient lowpass filter design for biomedical applications. 2010 IEEE biomedical circuits and systems conference, pp 130–133
5. Zhao Y, Yang Y, Kyas M (2011) Comparing centralized Kalman Filter schemes for indoor positioning in wireless sensor network. International conference on indoor positioning and indoor navigation, 2011, pp 1–10
6. Pang J, Singh I (2011) Accelerometer based real-time remote detection and monitoring of hand motion. Lecture notes in engineering and computer science: proceedings of the world congress on engineering and computer science 2011, WCECS 2011, vol II, 19–21 Oct 2011, San Francisco, USA, pp 744–747. http://www.iaeng.org/publication/WCECS2011/WCECS2011_pp744-747.pdf
7. Kalman RE (1960) A new approach to linear filtering and prediction problems 1. Trans ASME J Basic Eng 82(Series D):35–45