Hindawi
Mobile Information Systems
Volume 2021, Article ID 4741912, 10 pages
https://doi.org/10.1155/2021/4741912

Research Article

Equestrian Sports Posture Information Detection and Information Service Resource Aggregation System Based on Mobile Edge Computing

Zhong Wu[1] and Chuan Zhou[2]

[1]*Physical Education School, Wuhan Business University, Wuhan 430056, Hubei, China*
[2]*Institute of Mechanical Engineering, Wuhan Institute of Shipbuilding Technology, Wuhan 430062, Hubei, China*

Correspondence should be addressed to Chuan Zhou; 20150459@wbu.edu.cn

Received 19 May 2021; Revised 10 June 2021; Accepted 30 June 2021; Published 9 July 2021

Academic Editor: Sang-Bing Tsai

Equestrianism is a special sport. Equestrian sports have become popular in people's lives and have also developed rapidly in our country, becoming a trend. This paper aims to study the equestrian sports posture information detection and information service resource aggregation system based on mobile edge computing. The research of equestrian sports can greatly promote future development. For this reason, we use the power of science and technology to study the equestrian sports posture and make an equestrian sports posture information detection system. The system is designed to feed back the riding posture in real time and can provide equestrian athletes with a more scientific and healthier equestrian training posture. In order to fully verify the practicability of the system, at the end of the forum, the performance of the entire monitoring platform was tested, and three tests were designed, respectively: packet loss rate test, driving test, and riding test. The driving test proves the validity of data transmission, and the riding test proves the practicability of the monitoring platform's functions. The test results show that the maximum data packet loss rate is 0.6%, which can fully confirm that the monitoring platform has sufficient technical practicability, no problems have been found in practical applications, and it has good market and application prospects.

1. Introduction

With the improvement of people's living standards and the continuous progress of science and technology, more and more applications of posture detection systems appear in everyone's sight. The attitude detection system refers to measuring the information of each sensor; analyzing the position, speed, and attitude of the carrier and other related parameter information through data processing; and providing it to the back end detection and control system, to realize the perception and control of the carrier [1].

Many mobile terminals (such as mobile phones, tablets, and smart cars) are becoming increasingly popular in modern society and life, and users' requirements for data transmission speed and service quality are showing an exponential increase. Mobile cloud computing transfers the task of data processing to the cloud and can no longer fully meet the needs of users for low latency and high quality of service [2, 3]. In order to meet the above requirements, some people propose to migrate from a nearby mobile network to a cloud server near the user for computing. In order to meet the above requirements, some people advocate moving the cloud server to the mobile edge close to the user for computing [4]. In order to meet the user's requirements for low latency, low energy consumption, and high quality of service, complex computing tasks can be directly offloaded to the edge server. Due to the continuous development and improvement of my country's mobile terminal software and hardware level and the gradual development and improvement of other mobile communication network technologies, users not only are the main demanders of data resources but also enable them to act as the main providers

of data resources and calculate their own data. Storage resources are directly provided to other mobile edge computing, making it an important part of mobile edge data computing [5]. This brand-new paradigm not only enriches the construction resources of edge servers more flexibly and diversely but also enables us to more effectively save and deploy fixed edge servers.

The emergence of mobile edge computing has also solved the problem of high latency. The NavShoe system designed by Hadiana and Ginanjar is a position tracking system. Its core is an inertial sensor, which is embedded in the shoe to determine the person's position in the shoe. The microprocessor is based on the ARM core. STM32 and MPU6000 MEMS 3-axis accelerometers can collect, measure, display, and store system motion curves. It also analyzes the measurement data, has high accuracy and excellent user interface, and has a strong practical application background and practical value. But in fact, such microprocessors are difficult to produce and lack large-scale practicability [6]. The Internet of Things (IoT) is the development of the Internet, which aims to perceive, collect, analyze, and distribute data through IoT devices that constitute its core components. In order to save resources in such applications, effective node deployment and data aggregation techniques can be used. Zhang introduced the design and modeling of node deployment and data aggregation in Edge-IoT applications, as well as homogeneous and heterogeneous network solutions for smart agriculture. But only the method was proposed, and it was not applied in practice [7]. Kazimierski developed a motion human signal detection algorithm based on the autoregressive model during his more than ten years of research [8]. The simulation results show that this method can effectively reduce the false alarm rate, but it needs further verification in the actual system. The platform provides functions such as video image acquisition, moving target detection, target segmentation, target scale, and contour feature extraction [9, 10]. This improved template matching method can be used to identify human facial images, but it cannot be widely used on animals and other moving objects [11].

The innovation of this article is as follows: (1) the ARM core microprocessor is used and the software is written on this basis to create an equestrian posture information detection system based on mobile edge computing, which realizes the real-time detection of the equestrian athlete's posture and verifies the feasibility of this system through a series of testing methods. (2) The design cost of equestrian sports in this article is low, the selected environment is relatively good, the volume is small and stable, and it can meet the market demand.

2. Equestrian Sport Posture Information Detection Method

2.1. Image Sensor Detection. There is a certain connection between attitude movement and electrooptical stability tracking system, which can be explained by the angular velocity movement of the gyroscope in the inertial space. The combined operation of the feedback and servo control system can stabilize the visual axis of the device in the inertial space, and the axis is isolated from the interference caused by the movement of the carriage and ensures that the viewpoint remains unchanged or can be stabilized within a narrow range of stable accuracy, providing a stable processing platform for the image sensor [12]. Then, the image sensor is detected on this basis. The stabilized platform is the inner loop of the image tracking loop. The control amount generated by the stabilization loop is used as the input for the stabilization loop. Not only does the stabilized loop design stabilize the visual axis, but also the tracking loop provides a better-controlled object. The image processor forms a stable loop through a stable decentralized tracking controller and finally tracks the target on the platform [13].

By clicking on the armature voltage balance equation,

$$U_a + R_a I_a + L_a \frac{\mathrm{d}I_a}{\mathrm{d}t} + U_e. \tag{1}$$

Among them, a represents a known quantity and e represents another known quantity. The motor back-EMF equation is

$$U_e = K_b w_m. \tag{2}$$

Substitute (2) into (1) to get

$$U_a + R_a I_a + L_a \frac{\mathrm{d}I_a}{\mathrm{d}t} + K_b w_m. \tag{3}$$

From the torque balance equation of the motor,

$$M_m + J \frac{\mathrm{d}w_m}{\mathrm{d}t} = J w_m. \tag{4}$$

Among them, the torque output by the torque motor is

$$M_m = C_m I_a. \tag{5}$$

Substituting formula (5) into formula (4) can yield

$$C_m I_a = J w_m. \tag{6}$$

Pull changes to equations (3) and (6) to get

$$C_m I_a = J w_m s, \tag{7}$$

$$U_a = R_a I_a + L_a I_a s + K_b w_m. \tag{8}$$

From equations (7) and (8), we can obtain the transfer function from the armature voltage to the motor speed:

$$\frac{w_m(s)}{U_a(s)} = \frac{C_m}{(R_a + L_a s) \cdot Js + C_m K_b}. \tag{9}$$

From equation (9), the structure of the controlled object of the stable platform without mechanical resonance is shown in Figure 1.

$T_e = L_a/R_a$ is the electromagnetic time constant of the motor and the electromechanical time constant $T_m = JR_a/C_m K_b$, substituting it into equation (9) to simplify and get

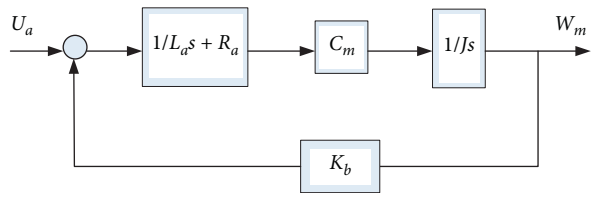

Figure 1: Structure diagram of the controlled object of the stable platform (excluding mechanical resonance).

$$\frac{w_m(s)}{U_a(s)} = \frac{1/K_b}{T_e T_m s^2 + T_m s + 1}. \tag{10}$$

2.2. Main Processor Motor Movement Principle. From the definition and physical meaning of the electromagnetic and electromechanical time constants of the motor, it can be seen that the electromagnetic transition time caused by the inductance of the motor itself is hundreds of times shorter than the electromechanical excess time caused by the moment of inertia. That is, $iT_m \gg T_e$; then, equation (10) can be transformed into

$$\frac{w_m(s)}{U_a(s)} \approx = \frac{1/K_b}{(T_m s + 1)(T_e s + 1)}. \tag{11}$$

It can be seen from equation (11) that, without considering factors such as mechanical resonance, viscous friction, damping characteristics, the characteristics of the controlled object of a one-dimensional gyro-stabilized platform can be regarded as a series of inertial links composed of two real poles. The pole with a lower frequency is mainly determined by the total moment of inertia of the platform, the total resistance of the armature, the motor torque coefficient, and the proportional coefficient of back electromotive force. The pole with a higher frequency is determined by the total inductance and total resistance of the armature [14]:

$$U_a = R_a I_a + L_a \frac{dI_a}{dt} + K_b W_m. \tag{12}$$

Equation (12) is the transfer characteristic of the torque generated by the motor to the speed of the motor shaft. From its transfer function, it can be seen that the system has a resonant frequency point and an antiresonant frequency point, as follows:

$$w_r = \frac{1}{2\pi}\sqrt{K_L\left(\frac{1}{J_m} + \frac{1}{J_L}\right)}, \tag{13}$$

$$w_r = \frac{1}{2\pi}\sqrt{\frac{K_L}{J_L}}. \tag{14}$$

3. Equestrian Sport Posture Information Detection Experiment

3.1. Hardware Design. This topic mainly collects posture data and motion data of riders and provides a database for research riding. The monitoring platform is mainly composed of three parts: 10 data collection nodes, 2 data receiving nodes, and 1 host. From the perspective of hardware design, the hardware circuit of each acquisition node is composed of a power supply module, a USB communication module, a single-chip microcomputer, and a wireless module [15]. From the perspective of software design, its main functions include sensor data collection, packaging, wireless communication, and USB data communication. The main functions of advanced software are serial communication, data conversion, display, and storage. The software and hardware work together to collect and store 3-axis angular velocity, 3-axis acceleration, and 3-axis magnetic data on the entire platform in real time, which provides convenience for future data analysis and algorithm research. Finally, perform performance testing on the entire monitoring platform. In this article, we will design three experiments separately: a packet loss rate test, a riding test, and a verification packet loss rate test [16].

The designed monitoring platform consists of three main parts: data collection node, data receiving node, and host. The data collection node of the system includes the main processor, sensor module, wireless transceiver, power cord, and interface, including the main processor, serial communication module, wireless transceiver module, power supply circuit, interface module, and the main computer. Accept raw data, convert and save, and view real-time curves. Collect, pack, and process 3-axis gyroscope data, 3-axis accelerometer data, and 3-axis magnetometer data and send them to the data receiving node at the data acquisition node. Then, the data receiving node is sent to the host, and the host performs data conversion and storage processing to facilitate subsequent data analysis and algorithm research [17].

Taking the SPI peripheral STM32L152, we will take data acquisition, wireless communication, serial communication software design, host software design, serial communication management, data conversion, storage, and data processing as examples to analyze the characteristics of serial communication in detail. Display according to the display curve and characteristics [18]. Among them, data acquisition software design, STM32L152 SPI peripherals, MPU9250 configuration and data acquisition, data packet format implementation and analysis, wireless communication software design, STM32L152 SPI peripherals and timers, nRF24L01+ configuration, application expansion burst mode, and 6-star network usage time interval method all cannot effectively solve the practical problems of wireless communication. In addition, the retransmission mechanism is used to increase the sensitivity of wireless reception and increase the reliability of wireless transmission. Design and use serial communication software STM32L152 to compare and analyze the advantages and disadvantages of data storage and transmission, and calculate the baud rate from the amount of data transmitted. The specific process is shown in Figure 2.

The inertial acceleration sensor is used to measure external inertial physical quantities, which mainly include linear acceleration and rotational angular acceleration.

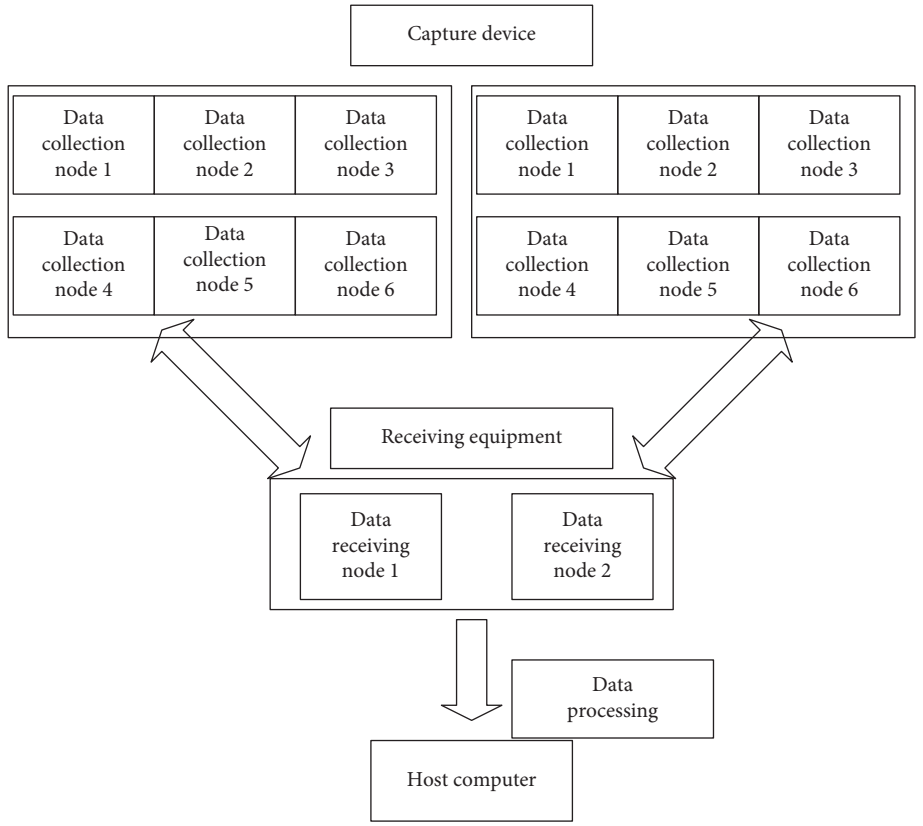

Figure 2: Monitoring platform structure block diagram.

Through filtering and integral operations, other parameters of the target, such as movement speed and angle, are obtained [19]. From a hardware point of view, it is mainly divided into main controllers, sensing equipment, mechanical key circuit, general power supply, and debugging equipment [20]. As can be seen from Figure 3, this is the overall hardware design of this article [21].

STM32F103RB is a high-performance MCU whose RISC core is ARM Cortex-M3. It is suitable for STM32F103xx medium-capacity expansion series, its power supply voltage is 2.0 V to 3.6 V, and the temperature range is between −40°C and +85°C [22]. At the same time, the energy-saving mode can ensure low power consumption requirements of the application. STM32F103xx medium-capacity series products provide 6 different packages and 36–100 pin packages. According to the main controller STM32MPU6000MPU6000 button infrared KEY and IRSPIGPIO power module POWER debugging interface of different packages, DEBUGSWDAD designed 6 kinds of posture based on MEMS motion sensor. Check the system and select different external configurations in the device. STM32F103RB belongs to an application product dedicated to 32-bit single-chip microcomputers with enhanced functions and lower cost in this series [23]. It has high performance, low power consumption, and rich and perfect instruction set, and the code is smaller than the existing 8-bit or 16-bit architecture [24]. The STM32F103RB works at a CPU frequency of up to 72 MHz. The on-chip peripherals of the STM32F103RB include built-in high-speed memory (128 K for flash memory, 20 K for SRAM), two APB buses, and three groups of 12-bit 21-channel

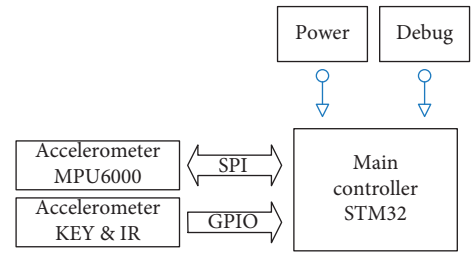

Figure 3: Overall hardware block diagram.

analog-to-digital converters (ADC), A 12-channel DMA controller, and multiple expansion I/O ports and peripherals connected to three general-purpose 16-bit timers, two sets of window watchdogs and independent watchdog timers, and standard and advanced communication interfaces: multiple up to 2 I2C and SPI interfaces, 3 USART interfaces, 1 USB interface, and 1 CAN interface. For details, please refer to the company's relevant technical information. Due to a large number of external configurations, STM32F103RB microcontrollers are very suitable for various practical applications, such as inverters, motor drives, application control, programmable logic controllers (PLC), printers, scanners, alarm systems, video intercoms, and HVAC Standby, providing a basis for selecting the main controller discussed in this white paper. providing a basis for selecting the main controller discussed in this white paperproviding a basis for selecting the main controller discussed in this white paper

Figure 4 shows the application program diagram of the equestrian sports monitoring platform. Some data collection nodes are located at different positions on the human body and the horse, measure the posture of the human body and the horse, and wirelessly send the posture data to the data receiving node. The data receiving node is connected to the host through the USB interface and sends the received data to the host for processing. The monitoring platform can measure 3-axis acceleration, 3-axis angular velocity, and 3-axis magnetic data to provide real-time feedback monitoring while driving.

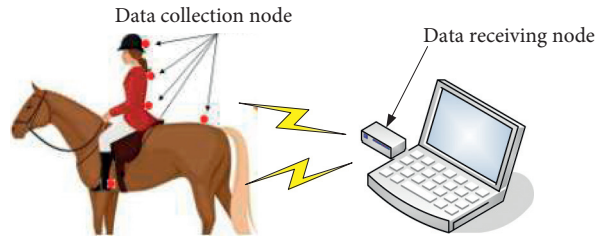

Figure 4: Monitoring platform application diagram.

3.2. Software Design. The software design of the monitoring platform includes two parts: embedded software and advanced computer software. According to the previous requirement analysis, two software functions are designed. It primarily completes the collection and transmission of inertial data and the handling of wireless channel conflicts. High-end computer software mainly uses the functions of receiving, converting, saving, curve, table, and so forth to display the received data in real time. Embedded software design mainly includes two parts: data acquisition and data reception. Figure 5 shows the software flow of the data acquisition node, and Figure 5 shows the software flow of the data receiving node. This article takes the STM32L152 chip as the core, uses the STM32L1 software library to perform external configuration and operations on the chip, ends the embedded software development, and discards the basic registers, respectively.

According to the above process, it is found that the data collection node mainly completes the inertial data collection and wireless communication functions, and the data receiving node mainly completes the wireless communication and serial communication functions. This article completes the functional analysis of embedded software from three aspects: data acquisition software, wireless communication software, and serial communication software. Inertial data collection is achieved using MPU9250. Both I2C and SPI have a data interface. Because the fastest communication rate of I2C is only 400 kHz, and the fastest communication rate of SPI is 20 MHz, the SPI method is chosen for data communication. STM32 L has two SPI interfaces, sending 8-bit and 16-bit data, respectively. The fastest SPI rate is half of the main frequency (16 MHz in this article), plus CPOL (clock polarity) and CPHA. Phase can be programmed to meet the SPI interface of different polarities. Since the command register of the MPU9250 can only support 1 MHz SPI communication speed at the fastest, the 1 MHz SPI communication speed is set to 1 MHz, and the SPI reading speed of the sensor data register and interrupt register is 20 MHz. Use 1 MHz rate communication when configuring MPU9250, and use 16 MHz rate communication when reading data. This improves the efficiency of data collection. DataGuide software settings: (1) initialization at power-up: SPI communication configuration, MPU9250 communication configuration: SPI external clock enabled, 4-wire pin configuration: SCK, MOSI, MISO pin push-pull multiplexed output, CS pin general I/O output; features configuration: open host mode, 8-bit data communication, clock polarity configuration, higher data priority, communication speed setting (1 MHz: MPU9250 configuration command, 16 MHz: MPU9250 read data); mpu 9250 configuration: accelerometer and gyroscope command configuration: select internal oscillator as clock source, update rate of data register of 200 Hz, gyroscope frequency of 1 KHz, accelerometer frequency of 1 KHz, low-pass filter of 5 Hz, acceleration. The measuring range of the meter is ±4 g, the gyroscope measuring range is ±2000 dps, the magnetometer is read out in I2C host mode, the magnetometer configuration is 16-bit AD conversion data, and the acquisition frequency is 100 Hz. MPU9250 data acquisition: based on the internal data register address of the MPU9250, the accelerometer, gyroscope, and magnetometer data can be read and stored in the Sent Data Package buffer for subsequent wireless transmission. Because the MPU9250 chip has a data AD conversion circuit, the hexadecimal inertia data can be directly obtained through the SPI interface. Table 1 shows the complete packet format.

In Table 1, the header and the end of the data packet occupy 1 byte each, allowing higher-level computers to split the data. When the advanced computer receives a large amount of data, it will determine the specific header and the end of the message. The byte length of each data packet determines whether the data packet has any bytes lost. The node number occupies 1 byte, which is used to distinguish different data collection nodes and promote higher classification and storage. A total of 18 bytes of the computer is collected from ACC_X_H to HZH as sensor data. The data of each axis contains two hexadecimal bytes. Both the accelerometer and the gyroscope are 2 bytes of data, the next data byte is 2 bytes, and the magnetometer is the upper and lower data bytes. The previous packet number is a byte from 0x00 to 0xff, which represents the number of packets sent to the current data. The time stamps are 2 bytes of TIM_L and TIM_H, respectively. The monitoring platform uses real-time wireless data transmission. In this case, data packets will inevitably be lost. Data collected by multiple collection nodes will not be aligned during data processing due to packet loss. Add a 2-byte time stamp, perform one-step data fusion on the data packet, and process the data that is difficult to download. This time stamp is sent uniformly from the data receiving node, rather than generated by the data collecting node itself. At the same time, it is convenient to adjust and evenly process all data collection nodes that collect tags.

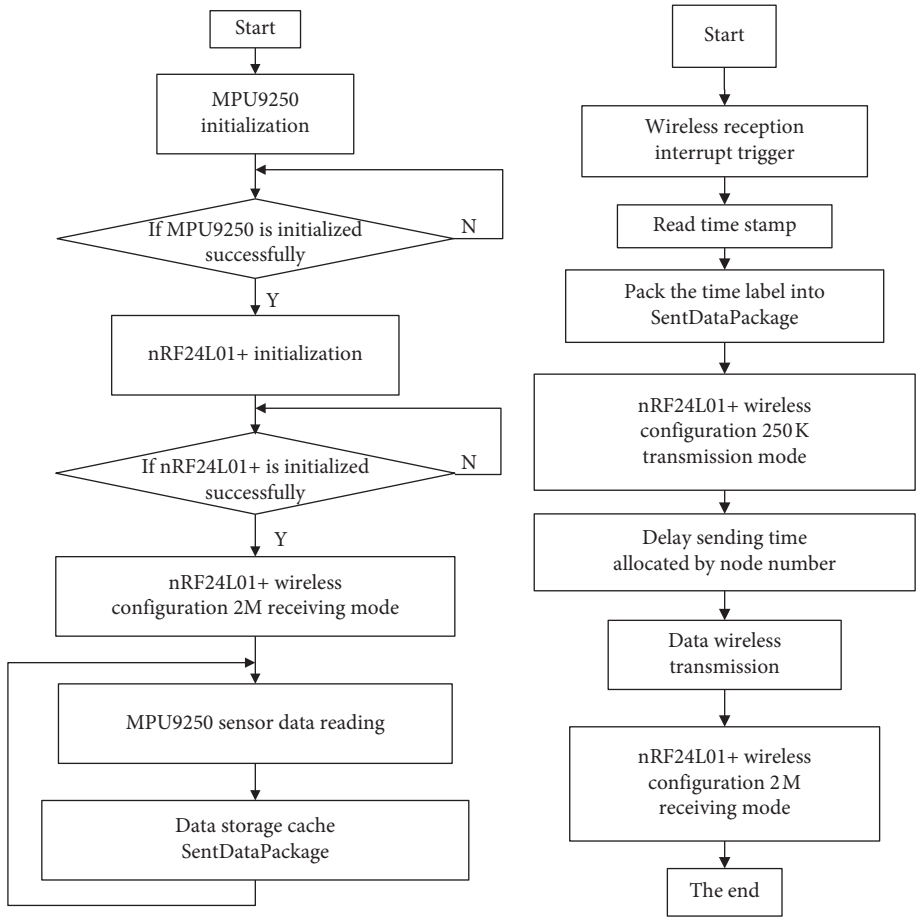

Figure 5: Software flowchart of data acquisition node.

Table 1: Packet format.

Beginning	Node number	Acceleration	Angular velocity	Magnetometer	Number	TIM_L	TIM_H	End
41	01	Date 1	Date 2	Date 3	8F	7E	5	45

After receiving the data, the data of multiple nodes are evenly stored in the receiving buffer string. As mentioned earlier, the data group starts at 41 and ends at 44 to make it easier for advanced computers to divide the data. The higher-level computers are started according to the group. Data is divided into three conditions: byte, end of data packet, and length of data packet. If all three restrictions are met at the same time, it is considered valid data. Otherwise, the data will be considered invalid and the data will be split again. After being divided into valid data, the physical data is converted according to the byte order of the packet. The calculation formula (15) is the acceleration conversion equation, the calculation formula (16) is the angular velocity conversion equation, and the calculation formula (17) is the magnetometer conversion equation. This is the sensitivity of the accelerometer, and $w\_$ is the sensitivity of the gyro. The sensitivity of this device is a magnetometer. Various MPU9250 range settings support various sensitivity values. The specific values are shown in Table 2.

$$a = \frac{N}{a} - \text{sensitivity} \times g, \tag{15}$$

$$w = \frac{N}{m} - \text{sensitivity} \times \frac{\pi}{180}, \tag{16}$$

$$m = N \times m - \text{sensitivity}. \tag{17}$$

N is composed of 2 bytes of data, which can be combined by bit operations. After the data conversion is completed, define the File Stream variable to create the file, define the file associated with the Stream Writer variable, call the Write Line() function to write the data into the file, and complete the data saving. All data will be saved. The user interface will be displayed in CSV format. The host in this article saves variables in real time instead of saving data after receiving all the data. This can avoid program blocking due to receiving too much buffered string data. After saving the data, you can delete the data by clicking the create or view data button or the delete folder button.

Table 2: Sensor sensitivity.

a_sensitivity	w_sensitivity	m_sensitivity
2048	16.7	0.5
4096	33.5	0.16
8192	65.5	
16384	134	

Table 3: Static test results.

Standing time (min)	Acceleration (msb)	Angular acceleration (msb)	Speed ($mm.s^{-1}$)	Angular velocity ($^{\circ}.s^{-1}$)
1	33	34	21	2
1.5	44	41	41	4
2	46	41	51	6
2.5	49	43	71	7.5
3	46	46	91	8.5
3.5	51	47	101	9.5

4. Equestrian Sports Posture Information Detection System

The system mainly includes signal acquisition, PCB processing, host analysis, display software, and other components. In certain testing and debugging processes, attention should be paid to separating software and hardware testing and debugging to facilitate the discovery and resolution of problems. In the debugging process, please pay attention to separate the functional modules of each part for independent testing, and finally conduct the overall test.

4.1. Static Analysis. In the process of numerical integration, it is easy to cause divergence and fluctuation of the integration curve. If you stop the system, the output of the system will be static at this time. For time integration, the static output changes with time. You can use a linear curve to compensate for this difference. The static curve correction strategy is theoretically effective because the system cannot be left unattended for long periods of time. The following is the static test result (msb, minimumsignificantbit, least significant bit) as shown in Table 3 and Figure 6.

4.2. Dynamics Test. When the system is placed in the actual measurement scene, the system output becomes dynamic output. If the dynamic output of the system tends to diverge, the algorithm will be disabled because it uses an effective filtering method. If the dynamic output of the system tends to be stable, the algorithm is effective, and the filtering is proven effective. The following are the results of the dynamic test shown in Table 4 and Figure 7.

It can be seen from the data in the figure that the posture display starts from a certain position after effective static data correction and dynamic filtering and finally returns to the approximate original position. The curve is smooth and complete, as far as I know. In addition, more obvious external vibrations are artificially added, and the attitude capture curve response is captured more clearly, thereby improving the dynamic characteristics. In general, this article can meet the design requirements within a higher accuracy range.

4.3. Horse Riding Test. This riding test is designed to measure the rider's backward posture, mainly to test the data collection status of the monitoring platform in actual application scenarios. The test was conducted at the Dalian Women's Cavalry Base under the guidance of professional coaches.

The detailed experiment process is as follows: the experimenter installs a data acquisition node in four locations, connects the data receiving node to the computer, turns on the host, and then receives the data. The receiver is centered and the experimenter can play. Data collection and storage have been completed for walking on a bicycle circular path within a distance of 7 meters (bicycle speed: slow, fast, slow, and fast); magnetometer calibration: each node of X, Y rotates around the z-axis to collect data.

This test is mainly to evaluate the status of platform data reception and the effectiveness of basic data. The 4 nodes receive data 19855, 19740, 19805, and 1985, respectively. The calculated maximum packet loss rate is 0.6%, and the packet loss rate is relatively high. At this time, the experimenter bends the horse's back forward to indicate the start of the experiment. Zone B is where the horse is slow and under the leadership. The coaches in Zone D jog, and the acceleration data is almost unchanged. Area E is the horse's fast running trot. At this time, the horse's jogging trot data is basically unchanged. The results show that the platform can collect riding data. You can feed back the original data. The exercise status was also compared with the videos taken in the experiment, which provided basic data on horseback riding therapy.

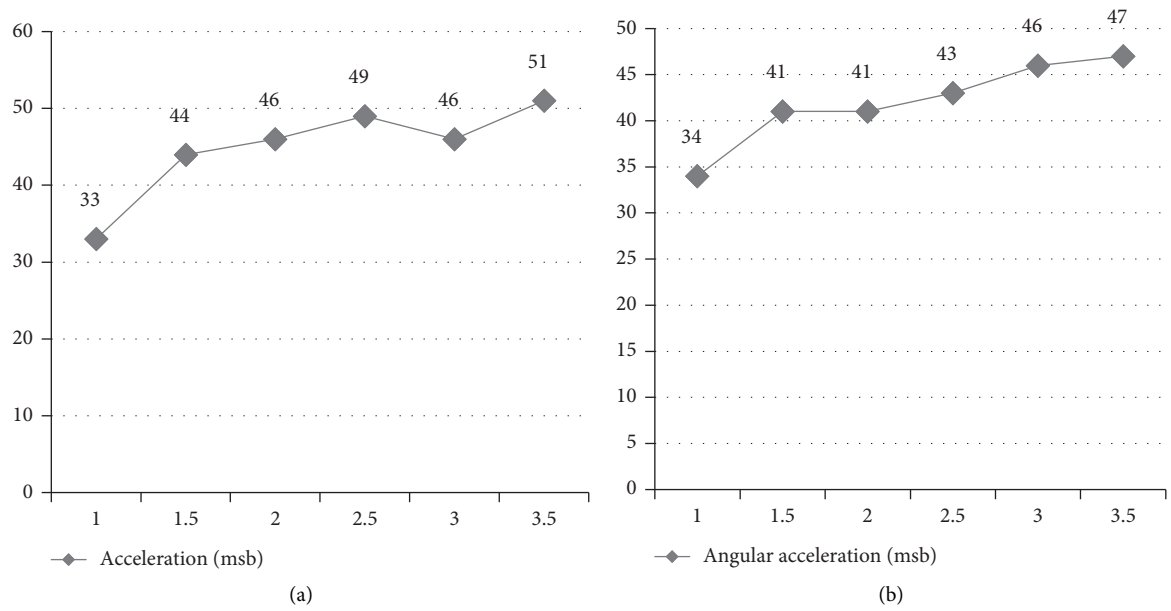

Figure 6: Changes in acceleration during static testing.

Table 4: Dynamic test results.

Standing time (min)	Acceleration (msb)	Angular acceleration (msb)	Speed (mm.s^{-1})	Angular velocity (°.s^{-1})
1	11	6	21	5
1.5	16	9	21	6
2	16	11	19	3
2.5	19	9	21	5
3	16	6	21	3
3.5	11	9	21	6

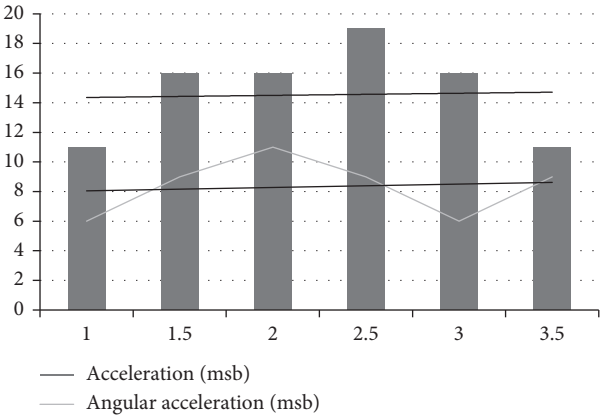

Figure 7: Dynamic test angular acceleration change.

5. Conclusion

A posture detection system based on the MEMS motion sensor is designed. According to the design goal, we first designed the entire system block diagram and then divided the entire work into modules based on the top-down idea; searched and summarized the data; and showed the design direction. It is constantly being modified. A large amount of data was queried, including the selection and comparison of accelerometers and the attitude fusion algorithms commonly used at home and abroad. In this process, the authors continue to develop and improve the ability to learn from reference materials and experience, combining hardware and software, appropriately emphasizing the engineering characteristics of the electronic and communication engineering fields, and at the same time, proposing new algorithm ideal. The workload is relatively high, including hardware design and debugging, software creation and testing, algorithm learning, and research. The monitoring

platform is small in size and easy to wear; does not affect the normal movement of the human body; has low power consumption and guaranteed working time; is convenient for practical use and wireless data transmission; and has no wired restrictions, wide range of uses, and strong riding ability. Although this design has reached the basic function of the design goal, due to time constraints and insufficient functions, there are still some areas that need to be improved. According to the original design ideas, the presenting talent has just completed. Due to the characteristics and architecture limitations of the ARM controller itself, compared with the DSP series processors, the ARM series STM32 processors used in this article have some defects in algorithm implementation. It is less efficient than DSP processors and allows you to try more data processing algorithms. In order to make the design more perfect, future work will be improved and make up for the above shortcomings.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest with any financial organizations regarding the material reported in this manuscript.

Acknowledgments

This work was supported by the Application Foundation Frontier Project (2020010601012294) and the School-Level Academic Team Project of Wuhan Business University (2018TD011).

References

[1] A. Bujari, S. Gaito, D. Maggiorini, C. E. Palazzi, and C. Quadri, "Delay tolerant networking over the metropolitan public transportation," *Mobile Information Systems*, vol. 2016, Article ID 8434109, 14 pages, 2016.

[2] Z. Wang, L. Jie, J. Wang et al., "Inertial sensor-based analysis of equestrian sports between beginner and professional riders under different horse gaits," *IEEE Transactions on Instrumentation and Measurement*, vol. 99, pp. 1–13, 2018.

[3] W. Kazimierski and M. Włodarczyk-Sielicka, "Technology of spatial data geometrical simplification in maritime mobile information system for coastal waters," *Polish Maritime Research*, vol. 23, no. 3, pp. 3–12, 2016.

[4] A. Jmc, B. Ka, A. Apcb et al., "Proposed injury thresholds for concussion in equestrian sports," *Journal of Science and Medicine in Sport*, vol. 23, no. 3, pp. 222–236, 2020.

[5] J. Zhang, J. Xiao, J. Wan et al., "A parallel strategy for convolutional neural network based on heterogeneous cluster for mobile information system," *Mobile Information Systems*, vol. 2017, Article ID 3824765, 12 pages, 2017.

[6] A. Hadiana and A. Ginanjar, "Designing interface of mobile parental information system based on users' perception using kansei engingeering," *Journal of Data Science and Its Applications*, vol. 1, no. 1, pp. 10–19, 2018.

[7] V. S. Kublanov, M. V. Babich, and A. Y. Dolganov, "Mobile hardware-information system for neuro-electrostimulation," *Mobile Information Systems*, vol. 2018, Article ID 2168307, 7 pages, 2018.

[8] G. Liu, S. Fei, Z. Yan, C.-H. Wu, S.-B. Tsai, and J. Zhang, "An empirical study on response to online customer reviews and E-commerce sales: from the mobile information system perspective," *Mobile Information Systems*, vol. 2020, Article ID 8864764, 12 pages, 2020.

[9] T. Koç, A. H. Turan, and A. Okursoy, "Acceptance and usage of a mobile information system in higher education: an empirical study with structural equation modeling," *The International Journal of Management Education*, vol. 14, no. 3, pp. 286–300, 2016.

[10] F. Mata, M. Torres-Ruiz, G. Guzmán et al., "A mobile information system based on crowd-sensed and official crime data for finding safe routes: a case study of Mexico city," *Mobile Information Systems*, vol. 2016, Article ID 8068209, 11 pages, 2016.

[11] C. N. Solis, F. Althaus, N. Basieux, and D. Burger, "Sudden death in sport and riding horses during and immediately after exercise: a case series," *Equine Veterinary Journal*, vol. 50, no. 5, pp. 644–648, 2018.

[12] V. Bachmann, B. von Salis, and A. Fürst, "Historical development of drug testing in Swiss equestrian sports," *Schweiz Arch Tierheilkd*, vol. 158, no. 4, pp. 259–265, 2016.

[13] C.-Y. Wang, H.-Y. Wei, and W.-T. Chen, "Resource block Allocation with carrier-aggregation: a strategy-proof auction design," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 3142–3155, 2016.

[14] Y. Nakamura, A. Moriguchi, M. Irie, T. Kinoshita, and T. Yamauchi, "Rule-based sensor data aggregation system for M2M gateways," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 12, pp. 2943–2955, 2016.

[15] Y. Cui, M. Kim, J. Jung et al., "Customized recommendation system based on rule matrix in SaaS aggregation service platform," *Advanced Science Letters*, vol. 23, no. 12, pp. 12769–12774, 2017.

[16] L. Niu, "Adaptive aggregation service for indoor location information using cloud," *Computer Speech & Language*, vol. 5, no. 4, pp. 341–362, 2016.

[17] F. Shi, D. Wu, D. I. Arkhipov et al., "ParkCrowd: reliable crowdsensing for aggregation and dissemination of parking space information," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–13, 2018.

[18] B. Sonkoly, R. Szabo, B. Nemeth et al., "5G applications from vision to reality: multi-operator orchestration," *IEEE Journal on Selected Areas in Communications*, vol. 99, p. 1, 2020.

[19] W. Dongliang, Z. Di, and D. Hongwei, "Digital library resources information integration system of self-help service," *Journal of Computational and Theoretical Nanoscience*, vol. 13, no. 12, pp. 9407–9411, 2016.

[20] G. N. Kouziokas, "Geospatial based information system development in public administration for sustainable development and planning in urban environment," *European Journal of Sustainable Development*, vol. 5, no. 4, pp. 347–352, 2016.

[21] J. A. Teresi, K. Ocepek-Welikson, M. Kleinman et al., "Psychometric properties and performance of the patient reported outcomes measurement information system (PROMIS) depression short forms in ethnically diverse groups," *Psychological Test and Assessment Modeling*, vol. 58, no. 1, pp. 141–181, 2016.

[22] K. Kalabokidis, A. Ager, M. Finney, N. Athanasis, P. Palaiologou, and C. Vasilakos, "AEGIS: a wildfire prevention and management information system," *Natural Hazards and Earth System Sciences*, vol. 16, no. 3, pp. 643–661, 2016.

[23] M. S. T. Ghatge and S. D. Bhoite, "Improving college working efficiency through design and development of a computerized information system (CIS)," *Siam Journal on Scientific Computing*, vol. 30, no. 4, pp. 1925–1948, 2016.

[24] J. Moon, Y. Choi, J. Kim et al., "An improvement of robust and efficient biometrics based password authentication scheme for telecare medicine information systems using extended chaotic maps," *Journal of Medical Systems*, vol. 40, no. 3, pp. 1–11, 2016.