Comprehensive Journal of Science
Volume (10), Issue (39), (March, 2026)
ISSN: 3014-6266

مجلة العلوم الشاملة
المجلد (10) العدد 39 مارس 2026
ردمد: 3014-6266

Design and Implementation of a Low-Cost Wireless Hand-Gesture-Controlled Wheelchair Using MPU6050 and NRF24L01

Hisham Aboulghasem Ali Esherwi
Lecturer at College of Computer Technology-Tripoli, Libya
hisham.esherwi@gmail.com

تاريخ الاستلام:2026/01/02 –تاريخ المراجعة: 2026/01/20– تاريخ القبول: 2026/01/29– تاريخ للنشر :2026/02/21

Abstract

This paper presents the design and implementation of a low-cost wireless hand-gesture-controlled wheelchair aimed at improving mobility and independence for individuals with physical disabilities. The proposed system uses an MPU6050 inertial measurement unit (IMU) embedded in a wearable glove to detect hand orientation and motion. The sensed data are processed by an Arduino Nano microcontroller and then transmitted wirelessly via an NRF24L01 module to a receiver unit installed on the wheelchair.

At the wheelchair side, a second Arduino Nano receives and interprets the transmitted commands, while an L298N motor driver controls two DC motors to perform directional movements, including forward, backward, left, right, and stop. The system architecture is designed with a strong focus on affordability, ease of use, and dependable performance.

Experimental testing under indoor conditions showed smooth system operation, stable wireless communication, accurate gesture recognition, and fast motor response. The findings indicate that IMU-based hand-gesture control can provide a practical and accessible alternative to conventional joystick-based wheelchair control, especially for users with limited upper-limb mobility.

Keywords: MPU6050, NRF24L01, Arduino Nano, L298N Motor Driver, Assistive Mobility, Wireless Wheelchair Control.

الملخص

يقدّم هذا البحث تصميمًا وتنفيذًا لنظام تحكم لاسلكي في الكرسي المتحرك باستخدام إيماءات اليد، بهدف تحسين الحركة وزيادة الاستقلالية لدى الأشخاص ذوي الإعاقات الجسدية. يعتمد النظام على حساس القصور الذاتي **MPU6050** المدمج داخل قفاز يُرتدى باليد، حيث يلتقط ميل اليد واتجاه حركتها بصورة دقيقة. ثم تُعالج بيانات الحساس بواسطة متحكم **Arduino Nano**، وتُرسل لاسلكيًا عبر وحدة **NRF24L01** إلى وحدة استقبال مثبتة على الكرسي المتحرك. في جهة الاستقبال، تقوم وحدة **Arduino Nano** أخرى باستقبال الأوامر وتفسيرها، بينما يتولى مشغل المحركات **L298N** التحكم في محركين كهربائيين لتنفيذ حركات الكرسي الأساسية، وهي: التقدم، التراجع، الانعطاف يمينًا، الانعطاف يسارًا، والتوقف. وقد تم تصميم النظام بالاعتماد على مكونات منخفضة التكلفة وسهلة الاستخدام، مع التركيز على تحقيق أداء موثوق وعملي.

أظهرت الاختبارات العملية داخل البيئات الداخلية أن النظام يعمل بسلاسة، ويوفّر اتصالًا لاسلكيًا مستقرًا، ودقة جيدة في تمييز الإيماءات، واستجابة سريعة في تشغيل المحركات. وتؤكد النتائج أن التحكم بالإيماءات المعتمد على الحساسات يمكن أن يكون بديلًا عمليًا وفعّالًا لوحدة التحكم التقليدية(**Joystick**) ، خاصةً للمستخدمين الذين يعانون من محدودية في حركة الأطراف العلوية.

1. Introduction

Assistive mobility technologies are essential for improving independence and quality of life for individuals with physical impairments. Most conventional electric wheelchairs depend on joystick-based control, which can be difficult—or even impossible—to use for people with limited upper-limb movement or motor control. This challenge has encouraged researchers and developers to explore alternative control interfaces that are more accessible and user-friendly.

Recent progress in embedded systems, inertial sensing, and wireless communication has made gesture-based control a promising option for assistive mobility applications. In particular, inertial measurement units such as the MPU6050 can accurately detect hand orientation and motion, while low-power wireless modules such as the NRF24L01 enable reliable communication between wearable controllers and wheelchair platforms. Together, these technologies make it possible to build intuitive systems without relying on expensive hardware or complex processing units.

Although earlier studies have proposed IMU-based, hybrid IMU–sEMG, and vision-based wheelchair control systems, many of these approaches remain limited by high cost, calibration complexity, or sensitivity to environmental conditions. To overcome these issues, this study proposes a low-cost wireless hand-gesture-controlled wheelchair based on MPU6050 and NRF24L01 modules. The proposed design replaces the traditional joystick with a wearable glove that maps simple hand gestures into wheelchair movement commands, offering a practical and accessible solution for users with upper-limb disabilities.

2. Literature Review

Gesture-based and sensor-driven wheelchair control systems have received considerable attention in recent years as researchers seek more accessible assistive mobility solutions.

Miftahussalam et al. (2023) demonstrated a reliable wheelchair control interface based on the MPU6050 sensor, highlighting the feasibility of low-cost embedded hardware for gesture recognition. Their work showed that a single inertial sensor can successfully convert hand orientation into directional movement commands.

Expanding on this direction, Iqbal et al. (2024) proposed an sEMG-based classification approach using support vector machines, achieving higher intent-recognition accuracy. However, the system depended on more expensive sensing components .

Similarly, Kundu et al. (2018) and Shahzad et al. (2019) combined IMU and sEMG signals to improve control robustness under varying limb positions. While these hybrid approaches improved motion smoothness and transition stability, they also introduced higher computational demands and more complex sensor calibration requirements.

In another approch, Haddoun et al. (2025) developed a head-movement-based control system using wearable inertial sensors. This approach is especially useful for users with severe upper-limb limitations, although it generally offers fewer control degrees and command flexibility .

More recently, Zhang et al. (2022) achieved fine-grained real-time gesture recognition by optimizing IMU data processing on embedded platforms. In addition, Branford et al. (2025) conducted a comprehensive inertial-sensor-based analysis of wheelchair user mobility, contributing datasets that can support the improvement of future assistive control algorithms. Han et al. (2024) also presented a broad review of intelligent wheelchair systems integrating IMU, vision, and IoT technologies, emphasizing the growing trend toward adaptive and context-aware assistive solutions.

Overall, previous studies can be grouped into two broad categories:

1. **Hybrid sensor-fusion systems**, which typically offer higher precision but require greater complexity and cost.
2. **Single-IMU systems**, which prioritize affordability, simplicity, and ease of use.

The present study follows the second direction, emphasizing a practical balance between reliability, simplicity, and low cost for real-world assistive applications.

3. System Architecture and Methodology

3.1 System Overview

The proposed system is composed of two main subsystems: a **transmitter unit** embedded in a gesture-detection glove and a **receiver unit** installed on the wheelchair. The transmitter

captures and interprets hand movements using an MPU6050 sensor, while the receiver translates the received commands into wheelchair motion through DC motors driven by an L298N motor driver. Communication between the two units is established wirelessly using NRF24L01 transceiver modules operating at **2.4 GHz**.

3.2 Hardware Components

The system is built from the following key hardware modules:

- **MPU6050 Inertial Measurement Unit (IMU):**
  Measures hand tilt and orientation using accelerometer and gyroscope data.
- **Arduino Nano (Transmitter):**
  Processes the IMU readings and sends encoded control commands through the NRF24L01 module.
- **NRF24L01 Wireless Module:**
  Provides low-power and reliable RF communication between the transmitter and receiver, with an indoor operating range of approximately **10 m**.
- **Arduino Nano (Receiver):**
  Receives and decodes the transmitted commands, then controls motor motion through the L298N driver.
- **L298N Motor Driver:**
  Converts low-power control signals into motor-driving signals and supports bidirectional control of two DC motors.
- **DC Motors:**
  Drive the rear wheels of the wheelchair and provide directional motion.
- **Power Supply:**
  A **9 V source** powers the control electronics, while a **7.4 V Li-ion battery pack** powers the drive motors.

3.3 Software Design

The software was developed in **Arduino IDE (C/C++)** and is organized into three main functional routines:

1. **Sensor Acquisition:**
   MPU6050 accelerometer and gyroscope readings are collected through the **I²C interface**.
2. **Gesture Classification:**
   Hand orientation is mapped to motion commands (forward, backward, left, right, stop) using predefined threshold-based conditions.
3. **Wireless Transmission and Motor Control:**
   The transmitter sends a single-character command through the NRF24L01 module, and the receiver decodes it to trigger the appropriate motor action.

Pseudocode for both transmitter and receiver logic is included in **Appendix A** and **Appendix B**. To improve stability and reduce noise, the IMU signals are filtered using a **complementary filter**, which helps maintain smooth control and low response latency.

3.4 Block Diagram

Figure 1 shows the overall communication and control architecture of the proposed system, including both transmitter and receiver sections.

The system is divided into two main sections:

- **Transmitter Section:**
  This section includes an Arduino Nano connected to the MPU6050 sensor, which detects the user’s hand orientation. The Arduino processes the sensor data and sends the corresponding control command wirelessly via the NRF24L01 module.
- **Receiver Section:**
  This section contains a second Arduino Nano connected to another NRF24L01

module. After receiving the command, the Arduino decodes it and activates the **L298N motor driver**, which controls two DC motors (Motor 1 and Motor 2) to perform forward, reverse, and turning movements.

This architecture provides a simple and efficient wireless control link between the gesture glove and the wheelchair drive system.

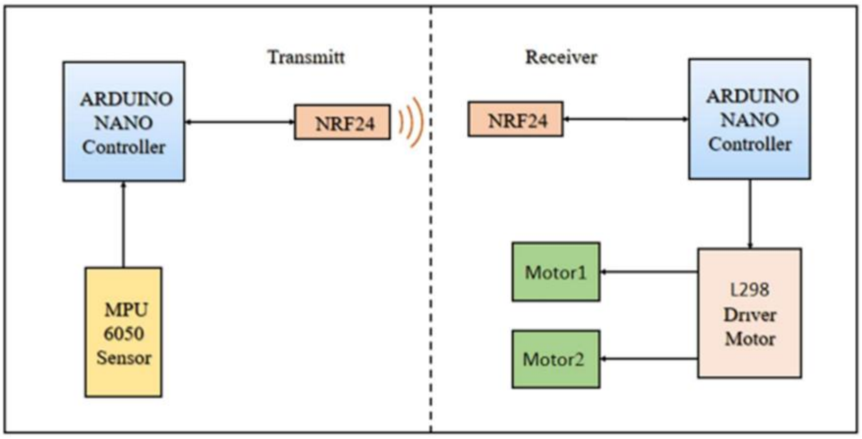

Figure 1. Block diagram of the transmitter and receiver sections.

4. Implementation

4.1 Hardware Implementation

The hardware implementation combines the designed components into two operational units: a transmitter unit and a receiver unit.

The **transmitter unit** (Fig. 2) consists of an **Arduino Nano**, an **MPU6050 sensor**, and an **NRF24L01 wireless module**. A **9 V power source** supplies the transmitter circuit. The Arduino reads and processes the MPU6050 data, then sends the corresponding gesture command wirelessly through the NRF24L01 module.

The **receiver unit** (Fig. 3) includes another **Arduino Nano**, an **NRF24L01 receiver module**, and an **L298N motor driver** connected to two DC motors. The motors are powered by **two 3.7 V Li-ion batteries**. The L298N driver enables bidirectional motor operation, allowing the wheelchair to move forward, reverse, and turn as required.

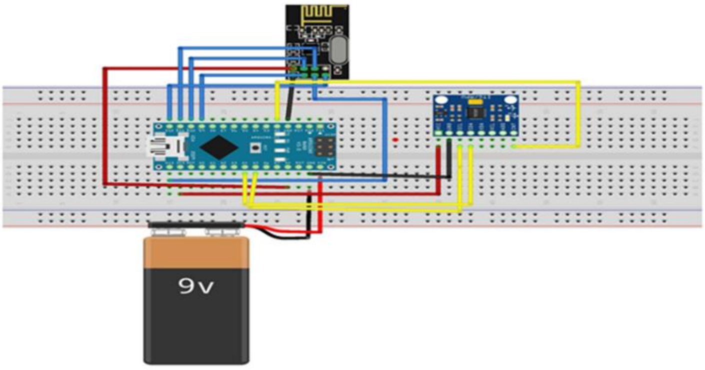

Fig. 2. Transmitter hardware implementation.

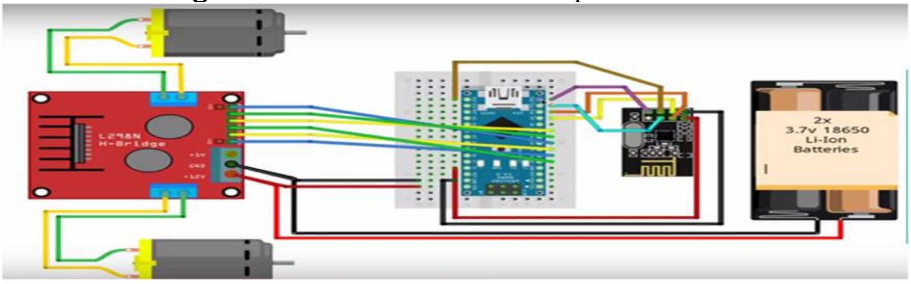

Fig. 3. Receiver hardware implementation.

4.2 Software Implementation

The software was developed using the **Arduino IDE** in **C/C++**.

On the transmitter side, the program continuously reads acceleration and gyroscope data from the MPU6050 sensor via the **I²C interface**, compares the readings with calibrated threshold values, and sends a corresponding character command (**“F”, “B”, “L”, “R”, “S”**) through the NRF24L01 module.

On the receiver side, the program continuously monitors incoming wireless data, decodes each received character, and generates the corresponding motor control signals for the **L298N driver**. This allows the system to execute the intended wheelchair movement in real time. The software logic was tested and debugged to ensure stable performance, reliable communication, and accurate gesture-to-motion mapping.

4.3 Experimental Setup and Prototype

The prototype was constructed using lightweight **acrylic sheets** mounted on a **two-wheel chassis**, as shown in **Fig. 4**. The electronic components were securely installed beneath the seat base to maintain mechanical stability and improve weight distribution.

The gesture-control glove, which integrates the **MPU6050 sensor**, was calibrated so that wrist movements corresponded to the main motion commands: **forward, backward, left, right, and stop**. Experimental tests demonstrated stable wireless communication between the transmitter and receiver units, along with smooth and accurate directional control of the wheelchair.

Fig. 4. Completed prototype of the hand-gesture-controlled wheelchair.

To further illustrate the transmitter subsystem, **Fig. 5** presents the wearable glove unit. In this design, the **MPU6050 sensor** and **NRF24L01 module** are integrated into a compact glove-based platform that captures hand orientation and transmits control commands wirelessly. The lightweight arrangement improves user comfort while maintaining stable and consistent gesture detection during operation.

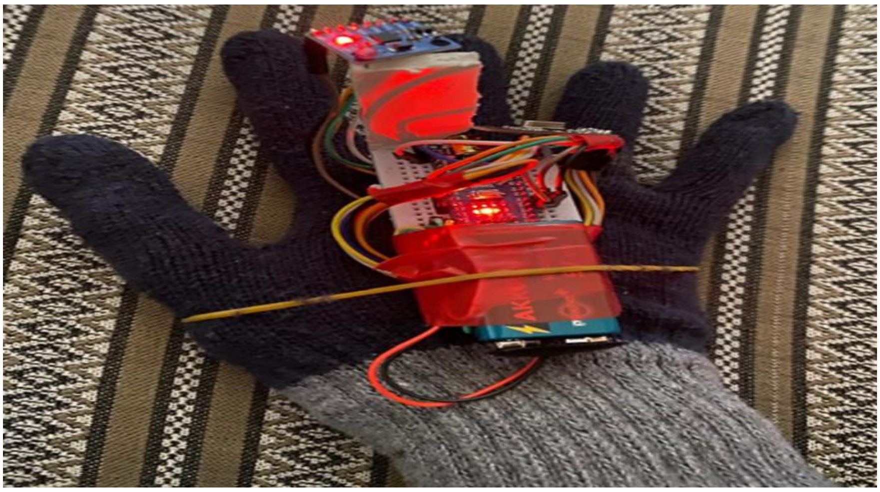

Fig. 5. Wearable glove-based transmitter unit incorporating MPU6050 and NRF24L01 modules.

5. Performance Results and Discussion

5.1 Performance Results

The implemented hand-gesture-controlled wheelchair system showed stable and consistent performance during laboratory testing. The wireless communication link between the transmitter and receiver remained reliable throughout the tested indoor range, with no noticeable interference from nearby electronic devices. In addition, the MPU6050 sensor successfully captured changes in hand orientation, while the Arduino Nano processed the sensor data in real time, resulting in fast and smooth motion response without perceptible delay.

To further assess the stability of the inertial measurement unit (IMU), the orientation angles—**Yaw, Pitch, and Roll**—were monitored during system startup and steady-state operation. As illustrated in **Fig. 6**, the sensor first passes through an automatic calibration stage during initialization, after which the orientation signals become stable with only minor fluctuations. The yaw signal converges quickly to its reference value and shows slight drift over time, whereas the pitch and roll signals settle smoothly after short transient variations. These results confirm that the IMU provides reliable and consistent orientation measurements suitable for real-time gesture recognition and wheelchair motion control.

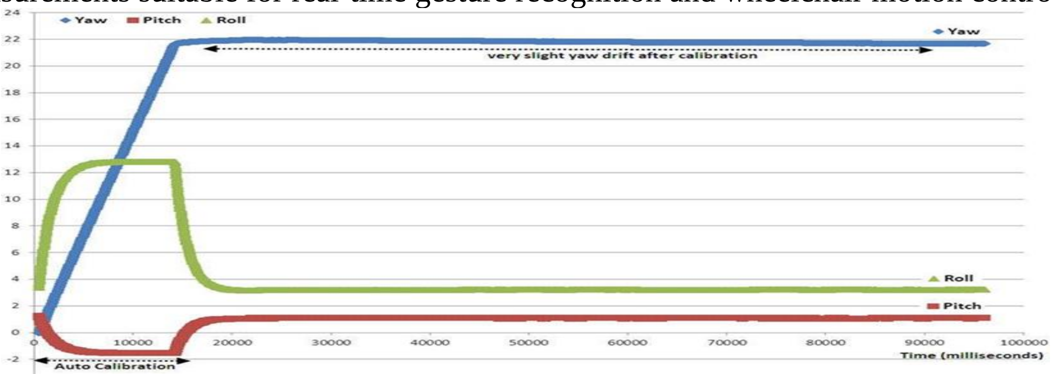

Fig. 6. IMU orientation response (Yaw, Pitch, and Roll) during calibration and steady-state operation.

The wheelchair responded accurately to the intended hand gestures—**forward, backward, left, right, and stop**—demonstrating effective coordination between the transmitter and receiver modules. Repeated trials also confirmed the robustness of the control logic, as no signal loss or irregular motor behavior was observed. Furthermore, separating the power supplies for the control circuit and motor actuation contributed to improved system stability and reduced electrical noise. Overall, these observations validate the capability of the

proposed system to provide smooth and intuitive wheelchair control using inertial sensing and low-power wireless communication.

5.2 Discussion

The experimental results suggest that the proposed system provides a practical and cost-effective solution for assistive mobility applications. Compared with earlier gesture-based and hybrid IMU–sEMG systems, such as those reported by **Miftahussalam et al. (2023)** and **Kundu et al. (2018)**, the present design offers comparable functional responsiveness while significantly reducing system complexity and hardware cost.

Unlike camera-based or vision-assisted control approaches (e.g., **Han et al., 2024**), the use of a single **MPU6050** sensor avoids the need for image processing, complex computation, and environmental calibration. This makes the proposed system more lightweight, portable, and easier to deploy in real-world settings. In addition, the **NRF24L01** module provides stable and low-power wireless communication that is well suited for indoor assistive environments, while the modular Arduino-based design simplifies maintenance, troubleshooting, and future upgrades.

From a usability perspective, the glove-based gesture interface offers a more intuitive control mechanism and may reduce the physical effort required from users with limited upper-limb mobility. The simplicity of the design also supports easy replication and adaptation for other assistive platforms, such as robotic arms, smart carts, or low-cost mobile assistive devices.

Although the prototype demonstrated reliable performance under controlled indoor conditions, several improvements can be considered in future work. These include integrating adaptive filtering techniques, adding feedback sensors for safer motion control, and incorporating IoT connectivity for real-time monitoring and data logging.

Overall, the proposed design successfully demonstrates that a wireless control architecture based on inertial sensing can deliver efficient, affordable, and user-friendly wheelchair control suitable for everyday assistive use.

6. Conclusion

This paper presented the design and implementation of a wireless hand-gesture-controlled wheelchair using the **MPU6050** sensor and **NRF24L01** wireless communication modules. The developed system demonstrated that a simple inertial-sensor-based approach, combined with low-power wireless transmission, can provide an intuitive, reliable, and cost-effective solution for assistive mobility.

Compared with hybrid **IMU–sEMG** approaches discussed in the literature, the proposed design avoids the need for multiple sensing modalities, which simplifies signal processing and reduces overall system complexity. Likewise, in contrast to higher-cost bioelectrical control systems such as those reported by **Iqbal et al. (2024)**, the presented model achieves practical responsiveness while remaining suitable for low-budget applications.

In addition, unlike camera- and vision-assisted wheelchair control methods (e.g., **Han et al., (2024**), the proposed system is not affected by lighting conditions or image-processing requirements. The MPU6050-based gesture interface enables direct and efficient mapping between hand motion and wheelchair commands, allowing fast and intuitive control without computational overhead. Furthermore, the use of open-source hardware platforms, particularly **Arduino Nano** controllers, improves the flexibility, maintainability, and scalability of the overall system.

Overall, both the experimental evaluation and the comparison with previous studies confirm that the proposed system offers a balanced trade-off between **simplicity, affordability, and functional performance**. The work represents a practical step toward intelligent assistive mobility devices for individuals with motor disabilities and shows that

dependable motion control can be achieved through efficient software integration and minimal hardware resources.

7. Future Work

Future development of the proposed hand-gesture-controlled wheelchair may focus on the following areas:

- **Advanced Filtering Algorithms**Integrating Kalman filtering or adaptive complementary filtering techniques to improve sensor accuracy and further reduce noise in MPU6050 measurements.
- **Closed-Loop Feedback Control**Adding wheel-speed or position sensors to implement closed-loop control for smoother motion, better tracking accuracy, and improved system stability.
- **IoT-Based Monitoring and Data Logging**Extending the system with Wi-Fi or GSM connectivity to support remote monitoring, performance tracking, and user-activity data logging.
- **Machine-Learning-Based Gesture Recognition**Applying lightweight machine-learning or neural-network models to enhance gesture classification and adapt the system to individual user movement patterns.
- **Power Optimization and Battery Management**Introducing intelligent power-management strategies and battery-monitoring circuits to increase operating time and improve energy efficiency.
- **Multimodal Assistive Interfaces**Combining hand-gesture control with additional input methods, such as voice commands or head-motion control, to accommodate users with different physical limitations.
- **Prototype Miniaturization and Ergonomic Enhancement**Improving glove comfort, optimizing sensor placement, and refining the wheelchair chassis structure to enhance usability, portability, and long-term user comfort.

Acknowledgment

The authors would like to express their sincere gratitude to the **College of Computer Technology, Libya**, for providing the facilities, technical support, and guidance that contributed to the successful completion of this research.

References

Branford, K. P.,Van Straten, M.G., Jahanian, O., Morrow, M. M. B. & Cain S. M. (2025). An inertial sensor-based comprehensive analysis of manual wheelchair user mobility during daily life in people with SCI. *PLOS ONE, 20*(8), e0323050. https://doi.org/10.1371/journal.pone.0323050

Haddoun, A., Djabri, D., Saidani, M. & Benbouzid, M. (2025). Development and evaluation of a head-controlled wheelchair using wearable inertial sensors. *Engineering Science and Technology, an International Journal*, *36*, Article in press. https://doi.org/10.1016/j.mex.2025.103485

Han, Y., Zhou, L., Jiang,W. & Wang G. (2024). Intelligent wheelchair human–robot interactive system: A review. *Journal of Mechanical Science and Technology, 38*. https://doi.org/10.1007/s12206-024-0731-7

Iqbal, H., Zheng, J., Chai, R. & Chandrasekaran, S. (2024). Electric powered wheelchair control using user-independent classification methods based on surface electromyography signals. *Medical & Biological Engineering & Computing, 62*, 167–182. https://doi.org/10.1007/s11517-023-02921-z

Kundu, A. S., Mazumder, O., Lenka, P. K., & Bhaumik, S. (2018). Hand gesture recognition-based omnidirectional wheelchair control using IMU and EMG sensors. *Journal of Intelligent & Robotic Systems, 91*, 529–541. https://doi.org/10.1007/s10846-017-0725-0

Miftahussalam, I., Julian, E. S., Prawiroredjo, K. & Djuana, E. (2023). Wheelchair control system with hand movement using accelerometer sensor. *Microelectronic Engineering, 271*, Article 112018. https://doi.org/10.1016/j.mee.2023.112018

Shahzad, W., Ayaz, Y., Khan, M. J., Naseer, N., & Khan, M. (2019). Enhanced performance for multi-forearm movement decoding using hybrid IMU/sEMG interface. *Frontiers in Neurorobotics, 13*, Article 43. https://doi.org/10.3389/fnbot.2019.00043

Zhang, D., Liao, Z., Xie, W., Wu, X., Xie, H. & Xiao, J. (2022). Fine-grained and real-time gesture recognition by using IMU data for assistive device control. *IEEE Transactions on Mobile Computing, 21*(12), 4421–4433. https://doi.org/10.1109/TMC.2021.3120475

Appendixes

Appendix A: transmitter code

```
#include <SPI.h>       //SPI library for communicate with the nRF24L01+
#include "RF24.h"      //The main library of the nRF24L01+
#include "Wire.h"      //For communicate
#include "I2Cdev.h"    //For communicate with MPU6050
#include "MPU6050.h"   //The main library of the MPU6050
MPU6050 mpu;
int16_t ax, ay, az;
int16_t gx, gy, gz;
int data[2];
RF24 radio(9,10);
const uint64_t pipe = 0xE8E8F0F0E1LL;
void setup(void){
  Serial.begin(9600);
  Wire.begin();
  mpu.initialize();             //Initialize the MPU object
  radio.begin();                //Start the nRF24 communicate
  radio.openWritingPipe(pipe);   //Sets the address of the receiver to which the program
will send data.}

void loop(void){
  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
  data[0] = map(ax, -17000, 17000, 300, 400 ); //Send X axis data
  data[1] = map(ay, -17000, 17000, 100, 200);  //Send Y axis data
  radio.write(data, sizeof(data));
}
```

Appendix B: receiver code

```
#include <nRF24L01.h>
#include <printf.h>
#include <RF24.h>
#include <RF24_config.h>
#include <SPI.h>      //SPI library for communicate with the nRF24L01+
#include "RF24.h"     //The main library of the nRF24L01+
const int enbA = 3;
const int enbB = 5;
const int IN1 = 2;    //Right Motor (-)
const int IN2 = 4;    //Right Motor (+)
const int IN3 = 7;    //Left Motor (+)
const int IN4 = 6;    //Right Motor (-)
int RightSpd = 130;
```

```
int LeftSpd = 130;
int data[2];
RF24 radio(9,10);
const uint64_t pipe = 0xE8E8F0F0E1LL;
void setup(){
  //Define the motor pins as OUTPUT
  pinMode(enbA, OUTPUT);
  pinMode(enbB, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(1, pipe);
  radio.startListening();
  }
void loop(){
  if (radio.available()){
    radio.read(data, sizeof(data));
    if(data[0] > 380){
      //forward
      analogWrite(enbA, RightSpd);
      analogWrite(enbB, LeftSpd);
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, LOW);
      digitalWrite(IN3, HIGH);
      digitalWrite(IN4, LOW);
    }

    if(data[0] < 310){
      //backward
      analogWrite(enbA, RightSpd);
      analogWrite(enbB, LeftSpd);
      digitalWrite(IN1, LOW);
      digitalWrite(IN2, HIGH);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, HIGH);
    }
    if(data[1] > 180){
      //left
      analogWrite(enbA, RightSpd);
      analogWrite(enbB, LeftSpd);
      digitalWrite(IN1, HIGH);
      digitalWrite(IN2, LOW);
      digitalWrite(IN3, LOW);
      digitalWrite(IN4, HIGH);
    }
    if(data[1] < 110){
      //right
```

```
    analogWrite(enbA, RightSpd);
    analogWrite(enbB, LeftSpd);
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
  }
  if(data[0] > 330 && data[0] < 360 && data[1] > 130 && data[1] < 160){
    //stop car
    analogWrite(enbA, 0);
    analogWrite(enbB, 0);
  }
}
```