

Robô controlado pelo movimento da mão

Guilherme Stiehl Ceroni, Roberto Luiz Debarba

Abstract. *Este artigo tem o objetivo de apresentar um robô controlado pelos movimentos da mão. Serão apresentados os passos para sua construção, com hardware e software. Também serão discutidas as aplicações do projeto, resultados atingidos e sugestões de extensão.*

1. Introdução

Este artigo tem o objetivo de apresentar um robô controlado pelos movimentos da mão. Serão apresentados os passos para sua construção, partindo do hardware ao software, contemplando os experimentos realizados.

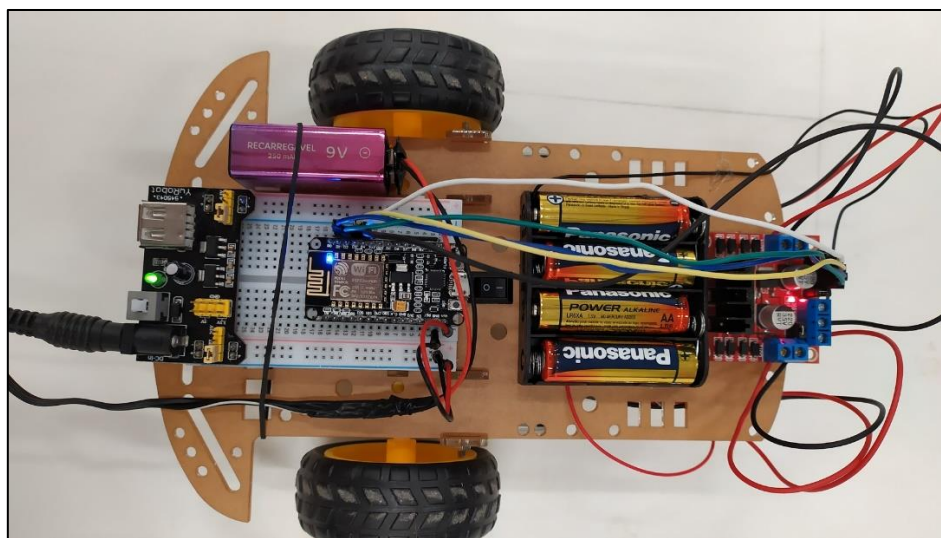
Se aplicado para o controle de brinquedos, como carrinhos de controle remoto ou outros veículos, esse controle por movimentos possui pouco valor. O controle apresenta precisão inferior aos controles tradicionais e cansa o usuário mais rápido. O principal objetivo do trabalho é testar e apresentar alternativas lúdicas para o desenvolvimento de exercícios na área da saúde, como fisioterapia. Outra utilidade de um controle de movimentos é a utilização para avaliação de performance física em diversas áreas.

2. Desenvolvimento do hardware

O hardware foi dividido em duas partes: robô e controle.

Na Imagem 1 é apresentado o robô montado. É possível visualizar a disposição das partes. Os componentes estão listados a seguir.

Imagem 1: Robô



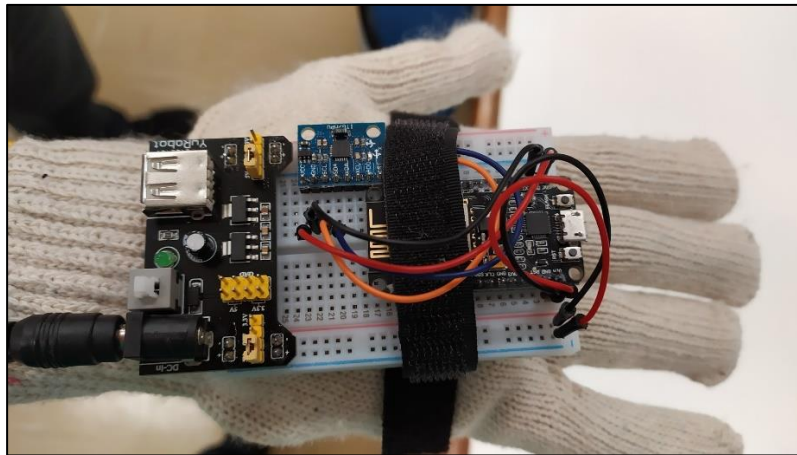
Fonte: Elaborador pelo autor

- Plataforma de acrílico que suporta todos os componentes. Esta pode ser substituída por outros materiais leves e resistentes, como MDF;
- Dois motores DC de 3 à 6V com redução e uma roda de 68mm cada;

- Roda boba universal;
- Switch on/off;
- Suporte para quatro pilhas AA;
- Protoboard de 400 pinos;
- Microcontrolador Esp8266 no módulo NodeMCU;
- Ponte H l298n para controle dos motores;
- Quatro pilhas AA para alimentação dos motores;
- Fonte de protoboard 3,3/5V para controle da tensão da bateria de 9V;
- Bateria 9V para alimentação do microcontrolador através do controlador de tensão.

Na Imagem 2 é apresentada a luva montada e vestida na mão. São mostradas todas as partes exceto a bateria de 9V, que preferimos anexar ao braço. Os componentes estão listados a seguir.

Imagem 2: Luva



Fonte: Elaborador pelo autor

- Microcontrolador Esp8266 no módulo NodeMCU;
- Protoboard de 400 pinos;
- Giroscópio 3 eixos Mpu-6050 Gy-521;
- Fonte protoboard 3,3/5V para controle da tensão da bateria de 9V;
- Bateria 9V para alimentação do microcontrolador através do controlador de tensão;
- Luva.

3. Desenvolvimento do software

O primeiro problema resolvido é a comunicação entre o robô e a luva. Com o objetivo de manter o custo baixo a melhor opção é a conexão Wi-Fi, já presente nos microcontroladores Esp8266. O robô deve ser ligado por primeiro e, ao iniciar, cria um hotspot Wi-Fi com SSID e senha fixos e aguarda a conexão de um cliente. Em seguida a luva deve ser ligada que, por sua parte, conecta no hotspot e agora pode enviar comandos para o IP 192.168.4.1, sempre atribuído ao robô.

O segundo problema tratado foi o protocolo de comunicação. Inicialmente optou-se pela transferência de mensagens via socket, sem nenhuma camada adicional. Com isso a comunicação e processamento seriam leves e rápidos, porém adicionava-se complexidade na interpretação das mensagens e tratamento da conexão para torná-la resiliente a falhas. Visando diminuir a complexidade do código e considerando que não seriam executadas operações que exigem grande processamento, o protocolo adotado foi o HTTP (Hypertext Transfer Protocol). Nele o tratamento de mensagens é mais simples e não é necessário tratar falhas de comunicação, pois a cada mensagem uma nova conexão é aberta. No Quadro 1 é apresentado o mapeamento das mensagens para as funções.

Quadro 1: Mapeamento dos comandos HTTP para funções

```
server.on("/", handleHttpRoot);
server.on("/left", left);
server.on("/right", right);
server.on("/front", front);
server.on("/back", back);

server.begin();
```

Fonte: Elaborador pelo autor

A terceira etapa é a obtenção da posição da luva através do giroscópio. É considerado um valor limiar para os eixos X e Y que definem a posição inicial para frente, trás, direita e esquerda. Para direita e esquerda foram usados os valores -10000 e 10000, respectivamente, e para a frente e trás foram usados os valores 8000 e -8000. Ao atingir algum desses valores é enviado um comando para o robô pela comunicação estabelecida.

A quarta e ultima etapa é a interpretação dos comandos enviados pela luva e execução dos movimentos através dos motores e ponte H. Inicialmente buscou-se controlar não apenas o estado de ligado e desligado dos motores, para avançar, voltar, virar a esquerda ou direita, como também a velocidade, recurso disponível no módulo l298n. Apesar do aumento de precisão dos movimentos, controlar a velocidade aumentaria a complexidade do código, então optou-se por manter a velocidade constante, sempre máxima, alimentando continuamente os pinos MA e MB do módulo através de jumpers.

4. Experimento

O experimento consiste em controlar o robô através de movimentos pela da luva de forma confortável e com precisão próxima à um controle tradicional.

Foi possível controlar o robô por aproximadamente 15 minutos sem cansar o braço, pulso ou mão, provando que os valores limiares de calibragem do giroscópio estão gerando o conforto necessário. Também foi possível controlar o robô através de corredores de aproximadamente 1,5 metros de largura atingindo o objetivo sem se chocar com as paredes, mostrando que os valores de calibragem também atendem ao objetivo de precisão.

5. Conclusão

O objetivo do trabalho de implementar e testar formas alternativas para a captura de movimentos foi concluído com êxito. O componente usado, um giroscópio de 3 eixos Mpu-6050 Gy-521 mostrou-se preciso e barato para o desenvolvimento deste trabalho e futuras extensões.

No quesito hardware esse projeto poderia obter diminuição de custo substituindo a protoboard, tanto do robô quanto da luva, por conexões diretas por jumper ou placas universais Pcb perfuradas. Os microcontroladores Esp8266 NodeMCU poderiam ser substituídos pelo Esp8266 fora do módulo, aumento a complexidade da construção, porém diminuindo o custo dos componentes. A fontes de protoboard também poderiam ser trocadas por reguladores de tensão, como o modelo L7805CV.

No software é possível aumentar a precisão do controle e resposta aos movimentos substituindo o valor limiar do giroscópio por um valor relativo que varia com a inclinação. Esse valor também deve ser interpretado de forma relativa usando velocidade variável. Por exemplo, ao inclinar o eixo X em 10° o robô vira à direita 10% da capacidade total de giro.