# HEAD DRIVING SYSTEM: INTERFACE HOMEM MÁQUINA PARA CONTROLE DE CADEIRAS DE RODAS ATRAVÉS DE MOVIMENTOS DA CABEÇA

Luiz Henrique Buzzi*, Marcelo Roberto Petry*

*INESC P&D Brasil
Departamento de Engenharias
Universidade Federal de Santa Catarina
Blumenau, SC, Brasil

Emails: buzzi@grad.ufsc.br, marcelo.petry@ufsc.br

**Abstract**— Powered wheelchairs can potentially provide more independence to people with reduced mobility. A significant part of this population, though, is not able to properly control the wheelchair through traditional joysticks. Therefore, it is important to develop new human-machine interfaces to improve the mobility of users with different degrees of impairment. This paper describes the development of the Head Driving System, a low-cost device that allows wheelchairs to be controlled through head movements. The device is composed of a microcontroller that collects data from an IMU and estimates the user's head orientation. This information is transmitted to the wheelchair controller, which generates the control references for each motor. The proposed system was tested in a virtual wheelchair implemented in a robotic simulator, demonstrating the system's features and capabilities.

**Keywords**— Wheelchair, Human-machine Interface, Reduced Mobility.

**Resumo**— Cadeiras de rodas motorizadas permitem que pessoas com dificuldades de locomoção se desloquem com maior independência. Entretanto, uma grande parcela desta população tem dificuldade de utilizar dispositivos tradicionais de controle, como joysticks manuais. Desta forma é importante que sejam desenvolvidas novas formas de interface homem-máquina que possibilitem que usuários com diferentes níveis de deficiências possam gozar de maior mobilidade. Este artigo descreve o desenvolvimento do Head Driving System, um dispositivo de baixo custo e tamanho reduzido que permite o controle da cadeira de rodas com movimentos da cabeça. O dispositivo possui um microcontrolador que coleta os dados de uma unidade de medida inercial e estima a orientação da cabeça do usuário. Esta informação é transmitida ao controlador da cadeira de rodas, o qual gera as referências de controle para cada um dos motores. O sistema proposto foi testado em uma cadeiras de rodas virtual implementada em um simulador robótico, demonstrando as funcionalidades e as capacidades do sistema.

**Palavras-chave**— Cadeira de Rodas; Mobilidade Reduzida; Interface Homem-máquina.

## 1 Introdução

Segundo dados do Instituto Brasileiro de Geografia e Estatística, 6,2% da população brasileira (ou aproximadamente 12,4 milhões de pessoas) possui algum tipo de deficiência (IBGE, 2014). O estudo mostra que 1,3% da população possui deficiência física, sendo que quase metade deste total (46,8%) possui grau intenso ou muito intenso de limitações, ou ainda não consegue realizar atividades habituais. Estima-se que 18,4% deste grupo frequentava algum serviço de reabilitação. A partir dos 30 anos, as proporções foram crescentes. Segundo esta fonte, 0,3% da população nasce com deficiência física, enquanto 1,0% adquire em decorrência de acidente ou doença.

Essa parcela da população tem ganho a atenção de organizações políticas e sociais para o desenvolvimento de soluções e produtos que se adequem a suas necessidades. A tendência é inserir as pessoas idosas e com deficiências motoras em grupos sociais e propiciar independência na realização de tarefas cotidianas.

Vários estudos têm mostrado que crianças e adultos podem se beneficiar substancialmente com o acesso a um meio de mobilidade independente, incluindo cadeiras de rodas elétricas, cadeiras de rodas manuais, scooters e caminhantes (Simpson, 2005). A independência da mobilidade aumenta as oportunidades profissionais e educacionais, reduz a dependência de cuidadores e familiares e promove sentimento de autossuficiência.

A solução não se trata da utilização de cadeiras de rodas motorizadas. É necessário o desenvolvimento de cadeiras de rodas com sistema modular, que permitam a integração de diferentes tipos de controle e que possam se adaptar a diferentes tipos de deficiência. A proposta de um sistema conjunto de protocolo aberto, permite que desenvolvedores apliquem seu conhecimento para o desenvolvimento de novas Interfaces Homem-Maquina (IHM). Este, permite que usuários utilizem o controle que mais se adeque às suas necessidades.

Muitos dispositivos presentes no mercado fornecem solução de controle de cadeiras de rodas a custos elevados. Podendo encontrar joystick manual a um custo entre R$1.350,00 e R$3.000,00.

Desta forma, este trabalho foca no desenvolvimento de um dispositivo de controle para cadeiras de rodas através de movimentos da cabeça. Utilizando uma unidade de medidas inerciais e um processador é desenvolvido uma IHM que permite que pessoas com mobilidade reduzida abaixo do pescoço consigam controlar os movimentos de sua

cadeira de rodas.

Este artigo está organizado da seguinte forma. A Seção 2 aborda a arquitetura de controle desenvolvida, separando as etapas do processo de tratamento de dados. Na Seção 3 é descrito o desenvolvimento do estudo de caso e os resultados obtidos. Por fim, na Seção 4 são apresentadas as conclusões e perspectivas para o trabalho realizado.

## 2 Arquitetura do Sistema

Esta seção trata da composição geral do Head Driving System (HDS), e será descrita em duas subseções: sistema de controle embarcado e driver do HDS aplicado ao controlador da cadeira de rodas.

O sistema de controle embarcado é responsável por coletar os dados de posição do HDS, fazer o tratamento das informações brutas coletadas dos sensores, estas recebidas na forma de aceleração e velocidade angular e transmitir as informações para o controlador da cadeira de rodas. Os elementos de hardware do sistema serão detalhados na subseção (2.1), e o algoritmo do sistema de controle na subseção (2.2). O driver, por sua vez, é responsável por fazer a conexão entre o sistema embarcado e o controlador. Este é um sistema instalado no controlador da cadeira de rodas, neste projeto é utilizado um computador portátil para fazer o controle do sistema. A arquitetura do software é descrita na subseção (2.3) Driver HDS.

### *2.1 Hardware do Sistema de Controle*

O hardware que compõe o presente projeto faz uso de componentes simples e facilmente encontrados no mercado nacional. Para estimar a orientação da cabeça do usuário a IMU utilizada foi a GY-80, que conta com um acelerômetro ADXL345 e um giroscópio HMC5883L. A IMU consiste em três acelerômetros e três giroscópios que fornecem três componentes de aceleração e três componentes de velocidade angular. Para estimar posição e orientação com as medidas da IMU, as acelerações e velocidades angulares devem ser integradas (Won et al., 2010).

Existem diversas vantagens em utilizar dois sensores neste projeto. Cada um possui qualidades e deficiências, entretanto, ao fundir os dados de ambos os sensores, é possível estimar informações mais precisas. O acelerômetro é capaz de fornecer o ângulo entre o vetor de aceleração sem acumulo de erro temporal, porém suas leituras possuem muito ruído. Qualquer variação do terreno ou movimentação da cadeira influenciam em sua leitura. O giroscópio consegue atribuir com precisão o ângulo formado com a movimentação da cabeça. Sua deficiência está vinculada ao acumulo de erro ao longo do tempo (Suprem et al., 2017; Luinge et al., 1999). Seu sistema é capaz de medir a velocidade angular instantânea, mas para este projeto é de interesse o deslocamento angular. Para isso é utilizado de artifício matemático que transforma a velocidade angular em deslocamento angular, porém este método carrega uma quantidade substancial de erro à medida que o tempo aumenta.

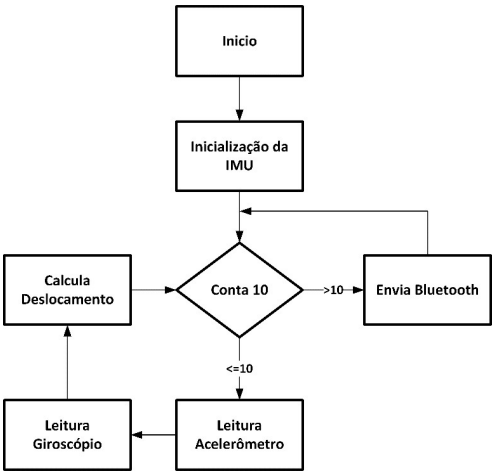


Figura 1: Fluxograma de Operação do HDS.

Os dados gerados pela IMU são coletados por um microcontrolador ATmega328 disposto em uma plataforma Arduino Nano. Este microcontrolador tem a tarefa de varrer a IMU a uma determinada frequência, transformar seus dados de velocidade angular e aceleração em deslocamento angular para ser aplicados no algoritmo de tratamento de dados descrito na seção 2.2. O microcontrolador irá calcular o posicionamento do HDS (cabeça do usuário) em relação ao seu ponto de calibração. Estes dados são enviados ao controlador da cadeira de rodas através de um transmissor Bluetooth (RS232 HC-05).

Na próxima subseção são apresentados aspectos relacionados à implementação do algoritmo de controle utilizado no sistema embarcado, com o intuito de demonstrar como são tratados os dados coletados da IMU e a saída para transmissor Bluetooth.

### *2.2 Algoritmo de Controle*

O sistema de controle consiste em um algoritmo separado em cinco etapas: inicialização da IMU e calibração dos sensores (Seção 2.2.1); obtenção e tratamento dos dados do acelerômetro (Seção 2.2.2); obtenção e tratamento dos dados do giroscópio (Seção 2.2.3); aplicação de método matemático que permite a integração dos dois sensores (Seção 2.2.4); envio dos dados para o controlador (Seção 2.2.5).

Na Figura 1 é ilustrado o fluxograma de operação do HDS. Na sequência é explicado o funcionamento de cada uma das cinco etapas de desenvolvimento do algoritmo.

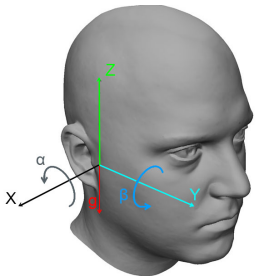


Figura 2: Orientação dos eixos do HDS.

### 2.2.1 Inicialização da IMU

A rotina de inicialização da IMU se procede em duas etapas. Primeiramente é necessário iniciar o acelerômetro. Para isto se faz necessário conhecer seu endereço de comunicação I2C e seus parâmetros de configuração. Este dispositivo já vem calibrado de fábrica (AnalogDevices, 2017).

A segunda etapa é a inicialização do giroscópio. Para inicializá-lo é necessário conhecer os endereços de comunicação I2C e seus parâmetros de configuração. Este dispositivo precisa ser calibrado via software, para isto é necessário o datasheet do fabricante, com parâmetros e o algoritmo para proceder. O algoritmo de calibração do giroscópio utilizado é construído seguindo as instruções da fabricante (Honeywell, 2010), que apresenta os algoritmos e constantes para calibração e obtenção dos dados.

### 2.2.2 Obtenção e Tratamento de Dados Acelerômetro

Os dados da IMU são coletados através da comunicação I2C. Os dados brutos são tratados conforme especificações do fabricante.

Os dados de aceleração linear nos eixos x, y e z, figura 2, retornados pelo acelerômetro são então convertidos para deslocamento angular. Tomando-se como referencial a aceleração gravitacional, as coordenadas angulares podem ser estimadas através do ângulo formado entre as acelerações nos três eixos:

$$\alpha_{ac} = \arctan\left(\frac{a_y}{\sqrt{{a_x}^2 + {a_z}^2}}\right) \quad (1)$$

$$\beta_{ac} = \arctan\left(\frac{a_x}{\sqrt{{a_y}^2 + {a_z}^2}}\right) \quad (2)$$

Onde $a_x$, $a_y$ e $a_z$ são as acelerações lineares nos eixos x, y e z, respectivamente, $\alpha$ é o angulo formado entre a gravidade e o eixo y e $\beta$ é o angulo formado entre a gravidade e o eixo x.

### 2.2.3 Obtenção e Tratamento de Dados Giroscópio

Os dados brutos são coletados da IMU através da comunicação I2C e tratados conforme especificações do fabricante. O giroscópio retorna valores de velocidade angular nos eixos x e y. Este sensor é mais estável pois sua leitura deriva da velocidade angular. É possível calcular o deslocamento angular utilizando uma equação simples:

$$\theta_n = \theta_{n-1} + \omega t \quad (3)$$

Onde $\theta_n$ é o deslocamento angular no instante, $\theta_{n-1}$ é o deslocamento angular no instante anterior, $\omega$ a velocidade angular e $t$ o tempo transcorrido entre as leituras.

A ideia é que se tomar a medida do giroscópio em intervalos de tempo regulares é possível estimar o deslocamento angular. O problema inerente deste método é a acumulação do erro de leitura, quanto maior o intervalo de tempo mais incerta a medida se torna.

### 2.2.4 Integração dos Sensores

Existem diversas metodologias que permitem a fusão de dados de sensores. Entretanto, dadas as limitações na capacidade de processamento do microcontrolador e nas necessidades do projeto, foi utilizado o filtro complementar.

O termo filtro complementar é utilizado como referência a um algoritmo que tenha o objetivo de fundir dados redundantes ou similares de diferentes sensores para estimar uma determinada variável. Este filtro opera no domínio da frequência e pode ser definido pelo uso de duas ou mais funções de transferência as quais complementam umas às outras (Higgins, 1975).

Para a aplicação neste projeto vai ser utilizada o seguinte modelo para o filtro complementar:

$$\alpha_n = A * (\alpha_{n-1} + \omega_x \Delta t) + (1 - A) * \alpha_{ac} \quad (4)$$

$$\beta_n = A * (\beta_{n-1} + \omega_y \Delta t) + (1 - A) * \beta_{ac} \quad (5)$$

Onde $\alpha_n$ e $\beta_n$ são os ângulo entre o eixo gravitacional e o dispositivo, $\omega_x$ e $\omega_y$ a velocidade angular, $\Delta t$ o tempo transcorrido entre as leituras, $\beta ac$ e $\alpha_{ac}$ são os ângulos obtidos pelo acelerômetro e A é uma constante de peso para o sensor, que varia de zero a um. Quanto maior for o valor de A maior vai ser a influência do giroscópio na estimação da posição angular.

Para este projeto foi utilizada uma configuração de A valendo 0,80. Este fator influencia na correção do erro do giroscópio e na redução do ruído de leitura do acelerômetro.

### 2.2.5 Transmissão de Dados

Continuamente são coletados os dados dos sensores e estimada a orientação da cabeça do usuário. A uma frequência de dez porcento da frequência de amostragem o sistema envia um pacote de dados para o transmissor Bluetooth, indicando para o driver qual o posicionamento do HDS em relação ao seu ponto de calibração.

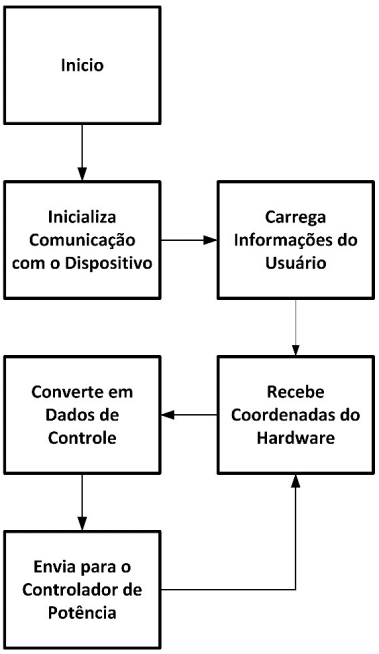


Figura 3: Fluxograma do Driver HDS.

### *2.3 Driver HDS*

O hardware do HDS é responsável por transmitir os ângulos de posicionamento do dispositivo em relação à origem. Esses ângulos devem ser tratados e convertidos em sinais para os motores da cadeira de rodas.

A função do driver está relacionada ao tratamento dos valores recebidos do hardware e armazenar os parâmetros do usuário, como mostra a Figura 3.

O driver do HDS é composto por cinco módulos: Inicialização da comunicação com o dispositivo (Seção 2.3.1); Informações do Usuário (Seção 2.3.2); Recebimento de coordenadas do hardware (Seção 2.3.3); Conversão dos dados em sinais de controle (Seção 2.3.4); Envio dos dados para o controlador de potência (Seção 2.3.5).

#### 2.3.1 Inicialização da Comunicação com o Hardware

O driver inicia comunicando-se com o dispositivo embarcado através de uma interface serial. Para inicializar o dispositivo ele limpa o buffer do sistema, garantindo uma ausência de dados na pilha, e envia um comando de transmissão.

Durante a inicialização o sistema é calibrado. A calibração ocorre no controlador, neste protótipo é utilizado um computador acoplado a cadeira. Através da tela e avisos sonoros, o driver solicita ao usuário que se mantenha estático em uma posição zero durante alguns instantes. Ao concluir sua calibração ele pausa a transmissão, porém o hardware continua seu processo de varredura, coletando os dados do posicionamento continuamente.

#### 2.3.2 Informações do Usuário

Cada usuário possui uma forma diferente de controlar a cadeira, diferentes posições zero, e principalmente diferentes máximos e mínimos de controle. O ponto zero ou ponto neutro é definido como a posição em que o usuário se sente confortável para não haver movimento da cadeira. Ao movimentar sua cabeça em alguma direção, ele irá atribuir algum comando. Os pontos máximos e mínimos permitem que o usuário determine quais são os extremos de comando:

- A posição que a cadeira vai assumir a máxima velocidade para frente;
- A posição em que a cadeira assume o valor máximo para trás;
- A posição que cadeira assume valor máximo de rotação para a esquerda;
- A posição que cadeira cadeira assume valor máximo de rotação para a direita.

Esses valores são determinados pelo usuário na hora do seu cadastro. O software recomenda que o usuário posicione a cabeça na posição de maior valor, lembrando-o que deve prevalecer uma boa visão e conforto na hora do controle. Esses dados são recuperados cada vez que o usuário utilizar o dispositivo.

#### 2.3.3 Recebimento de Dados do Hardware

A cada ciclo de processamento o driver aguarda a atualização dos dados provenientes do hardware. Durante este intervalo de tempo o driver permanece parado. Por se tratar de um sistema de controle em tempo real, o tempo de resposta é um fator fundamental para o controle. Como o computador consegue processar informações de forma mais rápida que o hardware empregado, é fundamental que ele permaneça aguardando a nova instrução.

Ao receber o dado o driver trata o pacote recebido, convertendo a string em variáveis que podem ser utilizadas para calcular os valores de controle da potência.

#### 2.3.4 Conversão dos Dados em Sinais de Controle

Os dados recebidos pelo hardware são coordenadas angulares brutas, esses dados precisam ser convertidos em dados de controle para os motores da cadeira de rodas. O mapeamento dos movimentos do usuário é importante pois determina o comportamento de resposta da cadeira de rodas para o usuário.

Visando a melhor adaptação do usuário ao sistema, (Faria et al., 2012) propôs três mapeamentos de controle (Algoritmo A, Algoritmo B e

Algoritmo C). Conforme o estudo, com esses algoritmos é possível melhorar a experiência do usuário dando a possibilidade de personalizar a direção conforme o mapeamento que melhor se enquadre no seu estilo de condução.

**Algoritmo A**

Este é o mapeamento mais simples. Consiste em decompor os ângulos obtidos em dois vetores normatizados X e Y que variam de -1 a 1 e aplicar em (6).

$$\begin{cases} R = y - x \\ L = y + x \end{cases} \quad (6)$$

Onde R é a potência do motor direito e L a potência do motor esquerdo no intervalo de 0 (zero) a 1 (um).

**Algoritmo B**

Neste mapeamento, a distância da cabeça para o ponto zero ($\rho$) é proporcional a velocidade máxima da roda e o ângulo ($\theta$) da cabeça relativa ao vetor (0,1) determina como a velocidade é distribuída para as rodas. Partindo do princípio que X e Y são dois vetores normatizados e que variam de -1 a 1, tem-se (7) e (8).

$$R = \begin{cases} \rho, if\ -\frac{\pi}{2} \leq \theta \leq 0 \\ -\rho, if\ \frac{\pi}{2} \leq \theta \leq \pi \\ \frac{\rho \cos(\theta + \pi/4)}{\cos(-\pi/4)}, if\ 0 < \theta < \frac{\pi}{2} \cup -\pi < \theta < -\frac{\pi}{2} \end{cases} \quad (7)$$

$$L = \begin{cases} \rho, if\ 0 \leq \theta \leq \frac{\pi}{2} \\ -\rho, if\ -\pi \leq \theta \leq -\frac{\pi}{2} \\ \frac{\rho \cos(\theta - \pi/4)}{\cos(-\pi/4)}, if\ -\frac{\pi}{2} < \theta < 0 \cup \frac{\pi}{2} < \theta < \pi \end{cases} \quad (8)$$

**Algoritmo C**

Este mapeamento é uma versão atualizada do algoritmo A. Partindo do princípio que X e Y são dois vetores normatizados e que variam de -1 a 1.

$$\begin{cases} R = y - nx \\ L = y + nx \end{cases} \quad (9)$$

$$\begin{cases} if\ \frac{x}{2} > 0.1, nx = 0.1 + (\frac{x}{2} - 0.1)/2 \\ if\ \frac{x}{2} < -0.1, nx = -0.1 + (\frac{x}{2} + 0.1)/2 \\ else\ nx = \frac{x}{2} \end{cases} \quad (10)$$

#### 2.3.5 Envio de Dados para o Controlador de Potência

Concluído o mapeamento da orientação da cabeça, o sinal de controle é enviado para o controlador de potência dos motores. Neste projeto o envio dos dados é feito pela API do simulador. Em uma cadeira de rodas real, entretanto, esta tarefa exigiria alguns cuidados. Segundo (Petry et al., 2013), a

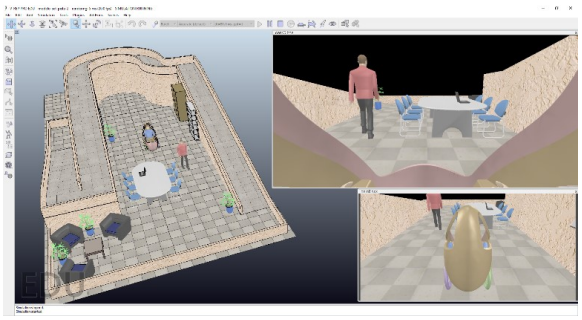

Figura 4: Simulador CDI.

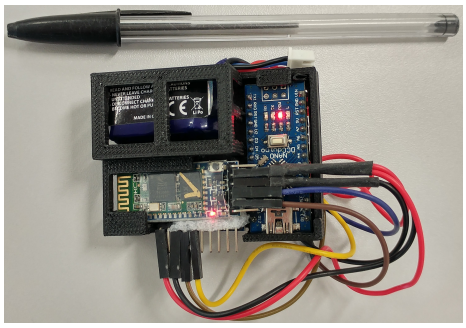

Figura 5: protótipo do sistema embarcado do Head Drivig System.

maioria dos controladores de potência utilizados em cadeiras de rodas utilizam versões proprietárias do protocolo CAN. Desta forma, enviar os sinais ao controlador exigiria: I)Conhecer o protocolo CAN utilizado; II) Abrir o joystick da cadeira de rodas e interceptar o barramento de controle; ou III) Utilizar controladores que possuam protocolos abertos.

## 3 Resultados

O ambiente de testes foi desenvolvido no simulador robótico V-REP (CoppeliaRobotics, 2013). No simulador foi desenvolvido um ambiente de teste que simula os principais aspectos e dificuldades do controle de uma cadeira de rodas em um ambiente comum. O simulador permite levantamento de colisões, estudo de aceitabilidade do mapeamento dos movimentos, aperfeiçoamento do sistema e treinamento dos usuários sem a necessidade de uma cadeira de rodas real.

Os testes realizados permitem descrever o funcionamento do sistema. Com o algoritmo utilizado não é possível notar atraso na transmissão das informações. O controle funciona de forma linear e em tempo real.

Os algoritmos de mapeamento alteram o modo de condução da cadeira de rodas. O algoritmo A demonstrou ser o mais intuitivo, visto que os comandos passados para o dispositivo são exatamente repassados para o controlador. O algoritmo B demonstrou requerer mais tempo para o aprendizado. Após absorvido, entretanto, se torna

mais eficiente pois suaviza os movimentos da cabeça e permite efetuar movimentos e trajetórias mais complexas. O algoritmo C é simples e tem um comportamento muito similar ao algoritmo A, porém acaba se tornando ruim por limitar a velocidade de rotação da cadeira. Nos extremos de comando de rotação a cadeira não atinge sua velocidade máxima.

O protótipo do HDS foi desenvolvido para ser fixado em armações de óculos. Qualquer um pode utilizar em seus óculos o complemento de controle (Figura 5). O protótipo conta com um hardware de tamanho reduzido e bateria própria, o que facilita sua fixação em diversos itens de vestuário.

## 4 Conclusões

O presente trabalho apresenta uma interface homem-máquina para o controle de Cadeiras de Rodas motorizadas através de componentes simples e de fácil acesso no mercado, como Arduino e IMU. A proposta permite que cadeiras de rodas motorizadas sem nenhum tipo de controle inteligente agreguem este dispositivo de baixo custo, possibilitando o controle para usuários com diversos tipos de deficiência física.

Com o objetivo de validar o protótipo implementado foi desenvolvido um ambiente virtual em um simulador robótico. Facilita o desenvolvimento do projeto assim como o treinamento dos futuros usuários. Com o foco na inclusão social, o dispositivo desenvolvido possui tamanho reduzido e pode facilmente ser fixado nos óculos dos usuários.

Com a conclusão do desenvolvimento do protótipo, serão realizados experimentos com pessoas sem comprometimento motor, com o intuito de aprimorar o sistema e encontrar eventuais falhas. Numa etapa seguinte, o sistema deverá ser reduzido, saindo da versão beta e sendo montado em uma plataforma singular, contendo apenas os componentes necessários para operação. Em campo poderão ser feitos testes com os usuários finais, pessoas com deficiências e idosos com dificuldades de locomoção. Os testes devem ser conduzidos em ambiente controlado, com o objetivo de coletar dados do usuário final e desenvolver um produto que possa ser lançado no mercado.

## Agradecimentos

Este trabalho é financiado pelo INESC P&D Brasil e pelo CNPq através do projeto "Cadeira de Rodas Inteligente: Design de Interface Multimodal de baixo custo" (edital Universal MCTI/CNPq Nº 14/2014) . O primeiro autor gostaria de agradecer ao CNPq pela concessão de sua Bolsa de Iniciação Científica (PIBIC/CNPq/UFSC – 2015/2016).

## Referências

AnalogDevices (2017). *Three-Axis Digital Accelerometer, ADXL345 datasheet*. Rev. E.

CoppeliaRobotics (2013). V-rep, *Coppelia Robotics GmbH, Zurich, Switzerland* .

Faria, B. M., Ferreira, L., Reis, L. P., Lau, N., Petry, M. and Couto, J. (2012). Manual control for driving an intelligent wheelchair: A comparative study of joystick mapping methods, *environment* **17**: 18.

Higgins, W. T. (1975). A comparison of complementary and kalman filtering, *IEEE Transactions on Aerospace and Electronic Systems* (3): 321–325.

Honeywell (2010). *Three-Axis Digital Compass, HMC5883L datasheet*. Rev. B.

IBGE (2014). Pesquisa nacional de saúde 2013: percepção do estado de saúde, estilos de vida e doenças crônicas.

Luinge, H. J., Veltink, P. H. and Baten, C. T. (1999). Estimation of orientation with gyroscopes and accelerometers, *[Engineering in Medicine and Biology, 1999. 21st Annual Conference and the 1999 Annual Fall Meetring of the Biomedical Engineering Society] BMES/EMBS Conference, 1999. Proceedings of the First Joint*, Vol. 2, IEEE, pp. 844–vol.

Petry, M. R., Moreira, A. P., Faria, B. M. and Reis, L. P. (2013). Intellwheels: Intelligent wheelchair with user-centered design, *2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013)*, pp. 414–418.

Simpson, R. C. (2005). Smart wheelchairs: A literature review, *Journal Of Rehabilitation Research And Development* pp. 423–438.

Suprem, A., Deep, V. and Elarabi, T. (2017). Orientation and displacement detection for smartphone device based imus, *IEEE Access* **5**: 987–997.

Won, S.-h. P., Melek, W. W. and Golnaraghi, F. (2010). A kalman/particle filter-based position and orientation estimation method using a position sensor/inertial measurement unit hybrid system, *IEEE Transactions on Industrial Electronics* **57**(5): 1787–1798.