

Daniel Noriaki Kurosawa

**Teleoperação de braço robótico por  
rastreamento de movimentos e imersão visual**

São Paulo

2016



Daniel Noriaki Kurosawa

# **Teleoperação de braço robótico por rastreamento de movimentos e imersão visual**

Trabalho de Formatura apresentado ao Departamento de Engenharia de Telecomunicações e Controle da Escola Politécnica da Universidade de São Paulo para obtenção do Diploma de Engenheiro

Universidade de São Paulo - USP

Escola Politécnica

Engenharia Elétrica com ênfase em Automação e Controle

São Paulo

2016

Daniel Noriaki Kurosawa

Teleoperação de braço robótico por rastreamento de movimentos e imersão visual/  
Daniel Noriaki Kurosawa. – São Paulo, 2016-

38 p. : il. (algumas color.) ; 30 cm.

Trabalho de Conclusão de Curso – Universidade de São Paulo - USP

Escola Politécnica

Engenharia Elétrica com ênfase em Automação e Controle, 2016.

1. Engenharia 2. Engenharia de Automação e Controle I. Universidade de São Paulo. Escola Politécnica. Departamento de Telecomunicações e Controle

# Agradecimientos

Agradecimientos aqui.



*“In a dark place we find ourselves,  
and a little more knowledge lights our way.  
(Yoda, Episode 3: Revenge of the Sith)*





# Resumo

Segundo a (??), o resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento. A ordem e a extensão destes itens dependem do tipo de resumo (informativo ou indicativo) e do tratamento que cada item recebe no documento original. O resumo deve ser precedido da referência do documento, com exceção do resumo inserido no próprio documento. (...) As palavras-chave devem figurar logo abaixo do resumo, antecidas da expressão Palavras-chave:, separadas entre si por ponto e finalizadas também por ponto.

**Palavras-chave:** latex. abntex. editoração de texto.



# Abstract

This is the english abstract.

**Keywords:** latex. abntex. text editoration.



# Lista de ilustrações

|  |    |
|--|----|
| Figura 1 – Diagrama do projeto . . . . .             | 22 |
| Figura 2 – Diagrama de Gantt do sistema . . . . .    | 27 |
| Figura 3 – Kanban do projeto usando Trello . . . . . | 28 |



## Lista de tabelas





# Lista de Gráficos



# Lista de abreviaturas e siglas

|        |   |
|--------|---|
| ADC    | Analog Digital Converter                          |
| EAP    | Estrutura Analítica de Projeto                    |
| FPGA   | Field-Programmable Gate Array                     |
| GF     | Galois Field                                      |
| IP     | Internet Protocol                                 |
| LED    | Light Emitting Diode                              |
| MAC    | Media Access Control                              |
| RS     | Reed Solomon                                      |
| USB    | Universal Serial Bus                              |
| UART   | Universal Asynchronous Receiver/Transmitter       |
| FET    | Field Effect Transistor                           |
| FLP    | Fast Locking Pattern                              |
| TDP    | Topology Dependant Pattern                        |
| PHR    | PHY Header  |
| SHR    | Synchronization Header                            |
| PSDU   | PHY Service Data Unit                             |
| MOSFET | Metal Oxide Semiconductor Field Effect Transistor |
| MCS ID | M   |
| RLL    | Run Length Limited                                |
| DC     | Direct Current                                    |
| AC     | Alternating Current                               |
| TTL    | Transistor-Transistor Logic                       |
| VPAN   | Visible-Light Communication Personal Area Network |
| TCP    | Transmission Control Protocol                     |



# Sumário

|            |   |           |
|------------|---|-----------|
|            | <b>Introdução</b>                         | <b>21</b> |
| <b>1</b>   | <b>ESPECIFICAÇÕES</b>                     | <b>25</b> |
| <b>1.1</b> | <b>Visão Empresarial</b>                  | <b>25</b> |
| 1.1.1      | Monetização do projeto                    | 25        |
| <b>1.2</b> | <b>Visão de informação</b>                | <b>26</b> |
| 1.2.1      | Elementos de informação do sistema        | 26        |
| 1.2.2      | Manipulação de informações                | 26        |
| 1.2.3      | Fluxo de informações                      | 26        |
| <b>1.3</b> | <b>Visão Computacional</b>                | <b>26</b> |
| 1.3.1      | Requisitos                                | 26        |
| 1.3.1.1    | Requisitos funcionais                     | 26        |
| 1.3.1.2    | Requisitos não funcionais                 | 26        |
| <b>1.4</b> | <b>Visão da Engenharia/Infraestrutura</b> | <b>26</b> |
| 1.4.1      | Arquitetura do sistema                    | 26        |
| <b>1.5</b> | <b>Visão Tecnológica</b>                  | <b>26</b> |
| <b>2</b>   | <b>METODOLOGIA</b>                        | <b>27</b> |
| <b>2.1</b> | <b>Planejamento</b>                       | <b>27</b> |
| 2.1.1      | Trello                                    | 27        |
| 2.1.2      | Cronograma                                | 28        |
| 2.1.3      | Requisitos                                | 29        |
| 2.1.3.1    | Requisitos Funcionais                     | 29        |
| 2.1.3.1.1  | Requisitos de hardware                    | 29        |
| 2.1.3.1.2  | Requisitos de software                    | 29        |
| 2.1.3.2    | Requisitos Não-Funcionais                 | 29        |
| 2.1.3.2.1  | Conclusão                                 | 29        |
| <b>2.2</b> | <b>Software</b>                           | <b>29</b> |
| <b>3</b>   | <b>EXECUÇÃO</b>                           | <b>31</b> |
| <b>4</b>   | <b>CONCLUSÃO</b>                          | <b>35</b> |
|            | <b>REFERÊNCIAS</b>                        | <b>37</b> |



# Introdução

Em muitas situações de risco ([BRUMSON, 2011](#)) é preferível hoje o envio de robôs a humanos devido necessitamos interagir com um ambiente remoto e

Braços robóticos possuem uma ampla gama de aplicações, substituindo seres humanos em tarefas em locais de difícil acesso, repetitivas, ou de alta periculosidade, indo seu uso desde o desarmamento de bombas ao uso em missões espaciais ([CSA, 2011](#)), passando por processos industriais ([BRUMSON, 2011](#)).

Entretanto, muitas das abordagens para sua operação continuam complexas e anti-intuitivas necessitando do uso de artifícios como controles remotos ([ICOR, 2016](#)) e joysticks ([MITSANTISUK; KATSURA; OHISHI, 2012](#)).

Uma segunda linha de pensamento tenta se aproveitar de semelhanças físicas entre o corpo humano e o braço robótico e faz uso, por exemplo, de sensores ([ALEOTTI; SKOGLUND; DUCKETT, 2012](#)) e marcadores ([WANG et al., 2005](#)) presos ao corpo. Entretanto, essas abordagens podem se provar incomodas ou restritivas em relação à movimentação do usuário devido por exemplo, a ocultação de um marcador ou a rigidez de um sensor vestível. ([DU et al., 2012](#))

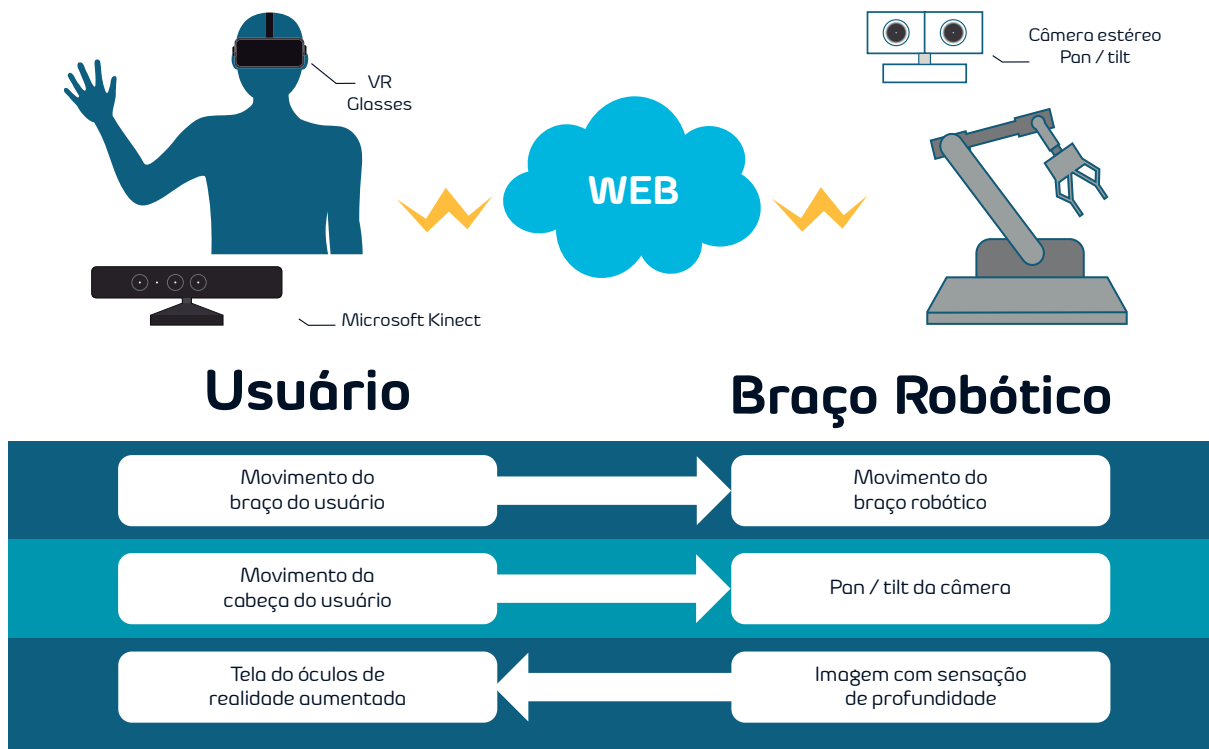
Uma solução para essas restrições é o uso de tecnologia de rastreamento visual sem marcadores, que permite que o usuário não mais se preocupe com as restrições de movimento impostas, tornando o controle mais natural. ([DU et al., 2012](#))

Paralelamente a operação do braço robótico, este trabalho estuda o uso de um sistema de imersão visual através do uso de um óculos realidade virtual por rastreamento dos movimentos da cabeça do usuário e um sistema de câmeras estereoscópicas ao projeto, com a finalidade de criar uma experiência imersiva ao usuário (.....)

## Motivação

Este projeto tem como antecedentes experiências com comunicação via luz visível. Criadas no contexto de uma disciplina de laboratório de processadores, as principais motivações foram a maior eficiência energética da VLC, bem como a diminuição de interferência com ondas de radiofrequência. No entanto, estas experiências esbarraram em desafios, principalmente relacionados à falta de um padrão de comunicação e de componentes adequados. Além disso, houve problemas relacionados à capacidade de processamento do microcontrolador utilizado, o que é consequência de um levantamento de requisitos incompleto.

Figura 1 – Diagrama do projeto



Fonte: Autor

Sendo assim, este projeto surge como um amadurecimento dos objetivos anteriores. Neste segundo momento, foi escolhida uma norma que atendesse às expectativas de implementar comunicação por luz visível de forma eficiente. O padrão IEEE 802.15.7 apresenta a vantagem de ter se completado em dezembro de 2011 e já ter se consolidado entre companhias e grupos industriais que formam o Consórcio Li-Fi. Além disso, o rigor metodológico é uma das preocupações centrais do novo trabalho, o que inclui um refinamento na maneira de definir os requisitos do sistema.

## Ideias iniciais e alternativas do projeto

Inicialmente, a pesquisa inicial tratou os seguintes temas:

- Geração de trajetória para braço robótico usando Microsoft Kinect;
- Captura de movimentos usando giroscópio;
- Algoritmos anti-drifting de giroscópio;
- Captura de imagens estereoscópicas em tempo real;
- Arduino;



- Raspberry Pi;
- Particle Photon;
- Internet of Things;

As ideias iniciais continham premissas que envolviam o uso de redes neurais para o controle do braço mecânico, utilizando um controlador ANFIS-PID para o controle dos motores, chegando a ser testado.

Posteriormente, em virtude de limitações de memória do hardware e a complexidade adicionada ao projeto, os algoritmos de redes neurais foram excluídos do escopo do projeto. O formato final é tratado na seção seguinte.

## Objetivo

Este estudo se propõe realizar o controle de um braço mecânico simplificando seu método de entrada de comandos sem elevar significativamente seu custo de projeto.

Colocado este objetivo, uma primeira preocupação do projeto é quanto a sua adaptabilidade. Para que o projeto possa ser aplicado a uma ampla gama de finalidades, o projeto foi subdividido em pequenos subprojetos, que podem ser utilizados com independentemente. Para isso será utilizado um ao realizar a comunicação em tempo real através da internet, introduzindo o conceito de Internet of Things (IoT) ao projeto. Serão levadas em considerações parâmetros como a robustez, taxa de transmissão de dados, custo da transmissão de dados, segurança do meio de comunicação, sendo necessária uma análise usando Architecture Tradeoff Analysis Method (ATAM) . Uma vez que o controle é realizado a distancia, este trabalho se propõe a implementar um sistema de câmeras imersivas estereoscópicas, em que a movimentação de duas câmeras instaladas próximas ao braço robótico será controlada pela movimentação da cabeça do usuário, sendo as imagens assim obtidas transmitidas uma à cada olho do usuário através de óculos visores, provocando a sensação de profundidade.



# 1 Especificações

Esta seção baseia-se nos 5 pontos fundamentais da ODP (Open Distributed Processing) para especificar o projeto a ser desenvolvido. Assim, divide-se este capítulo em: Visão Empresarial, de Informação, Computacional, de Engenharia/Infraestrutura e de Tecnologia.

## 1.1 Visão Empresarial

### 1.1.1 Monetização do projeto

O projeto a ser desenvolvido tem o intuito de analisar o comportamento do motorista ao dirigir, podendo assim ser utilizado como uma ferramenta de grande auxílio para companhias de seguro e monitoramento de frota. Companhias de seguros poderiam utilizar o sistema proposto para implementar um modelo de seguros para automóveis baseado em uma cobrança variável, segundo os hábitos de direção do motorista. Se um cliente é bastante cuidadoso no volante, o preço do seguro de seu carro seria menor. Contudo, seria necessário realizar algumas melhorias no projeto, como por exemplo: Impedir que o usuário desligue o módulo com intuito de mascarar os dados captados. Sem esta funcionalidade, um usuário mal-intencionado poderia desligar o aparelho quando estiver dirigindo de maneira perigosa; Enviar os dados sem o uso do celular do usuário, de forma a atuar independente do dispositivo. Empresas de monitoramento de frotas poderiam utilizar o sistema proposto para garantir a qualidade do comportamento de seus motoristas. Neste cenário, a melhoria necessária seria: Adicionar permissões de acesso aos usuários administradores da frota, que poderiam visualizar o histórico de toda a frota cadastrada. Desta maneira, com a implementação das funcionalidades especificadas acima, pode-se criar maneiras de criar renda e monetizar o sistema proposto.

O projeto desenvolvido tem o intuito de possibilitar a interação imersiva com um ambiente remoto através do uso de câmeras estereoscópicas e da operação de um braço robótico. Analisando o projeto, encontramos usos distintos para suas partes componentes em separado, assim como usos envolvendo as duas partes.

Para as câmeras, podemos Vigilância de ambientes, ao utilizar mais de um conjunto de câmeras, uma vez que o usuário poderia assinalar um endereço independente para cada um deles.

Jogos de realidade aumentada, ao possibilitar a movimentação baseada na cabeça do usuário, podemos

acesso a áreas de acesso restrito em exposições/zoológicos/aquários Áreas sensíveis a concentração de pessoas/visão de locais que normalmente não permitiriam o acesso ao público

Braço mecânico Programação de braços => necessária forma de gravar os dados

Conjunto Desarmamento de bombas Operação de reconhecimento em locais remotos Operações médicas => maior precisão do hardware/software

## 1.2 Visão de informação

### 1.2.1 Elementos de informação do sistema

### 1.2.2 Manipulação de informações

### 1.2.3 Fluxo de informações

## 1.3 Visão Computacional

### 1.3.1 Requisitos

#### 1.3.1.1 Requisitos funcionais

#### 1.3.1.2 Requisitos não funcionais

## 1.4 Visão da Engenharia/Infraestrutura

### 1.4.1 Arquitetura do sistema

## 1.5 Visão Tecnológica

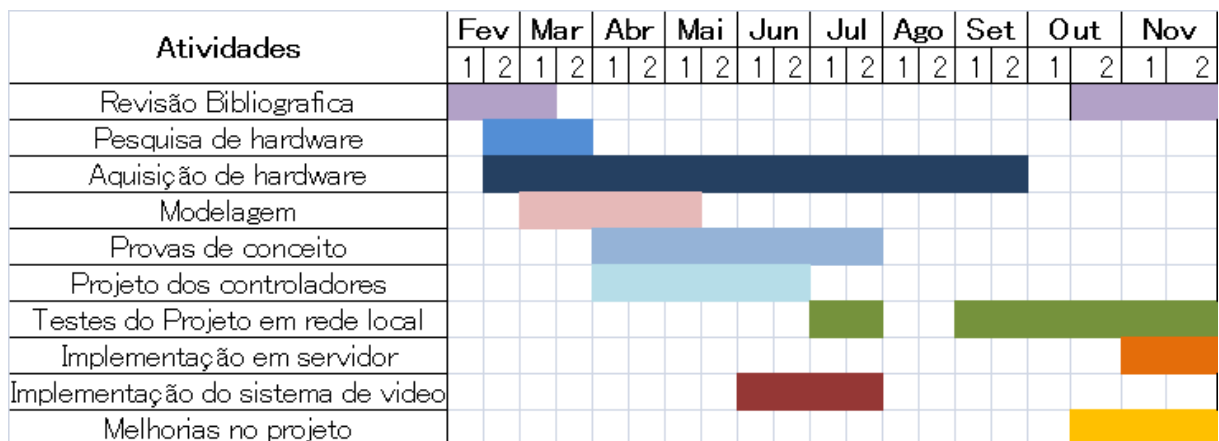
## 2 Metodologia

Este capítulo apresenta o método de trabalho adotado para a execução das atividades do projeto e divide-se em três partes: organização de tarefas, escolha de hardware e software utilizados.

### 2.1 Planejamento

Inicialmente, foi elaborado um diagrama de Gantt do sistema contendo as principais etapas do projeto (Figura 2) a partir do qual foram identificadas as tarefas a serem realizadas durante o projeto, que foram posteriormente transferidas para o Trello.

Figura 2 – Diagrama de Gantt do sistema



Fonte: Autor

#### 2.1.1 Trello

Trello é uma ferramenta de organização de projetos baseada nos quadros Kanban da metodologia ágil. Extremamente versátil, permite uma análise rápida da situação do projeto, das etapas cumpridas e do cronograma geral do projeto. No Trello, é possível criar “boards” que agrupam listas (“lists”) de tarefas que devem ser feitas. Cada tarefa é então representada por um “card”, criado dentro de uma “list”. As “lists” criadas dentro da “board” do projeto são:

**Backlog** : Tarefas que ainda precisam ser analisadas e aprovadas para serem colocadas na fase de atividades.

**Sprint** : Tarefas a serem realizadas após a conclusão das tarefas sendo trabalhadas atualmente.

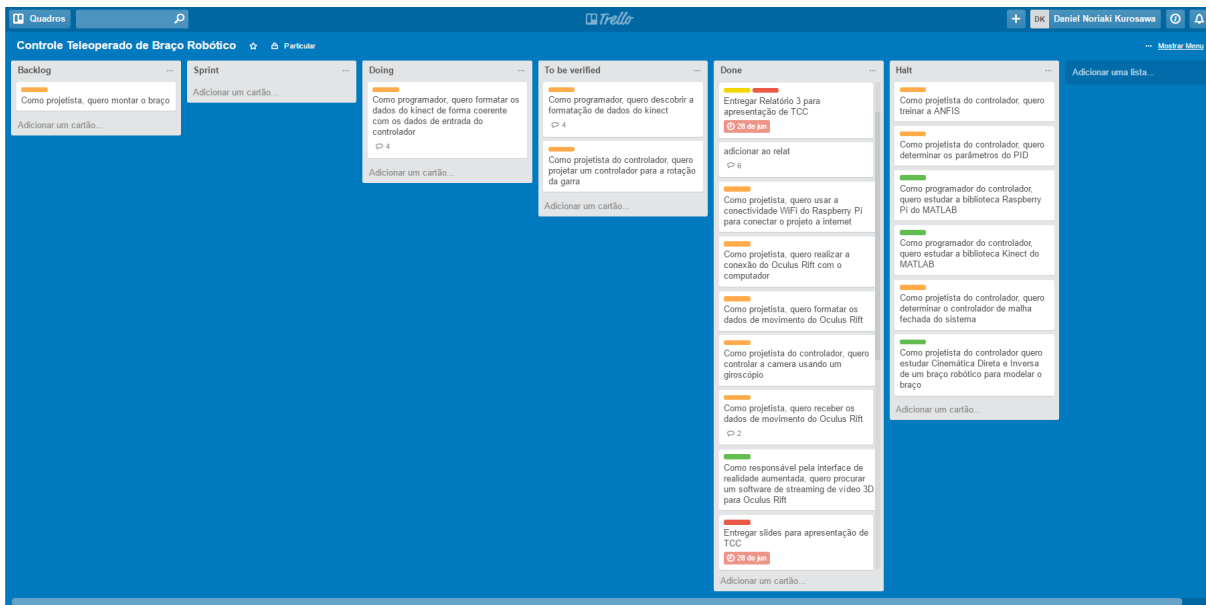
**Doing** : Tarefas em execução.

**To be verified** : Tarefas executadas a serem revisadas.

**Done** : Tarefas terminadas.

**Halt** : Tarefas suspensas/canceladas.

Figura 3 – Kanban do projeto usando Trello



Fonte: Autor

### 2.1.2 Cronograma

O cronograma do projeto foi planejado a partir dos pacotes de trabalho definidos na Estrutura Analítica de Projeto. Os participantes então fizeram reuniões para estimar o tempo, com a aplicação dessas estimativas a um calendário, considerando ainda as datas de entrega oficiais da disciplina. Com todas as estimativas feitas, o projeto tem a primeira e segunda fase com duração até o fim de Julho, com intuito de adiantar tanto a documentação quanto a implementação, como pode-se observar no diagrama de Gantt Figura 2 abaixo.

### 2.1.3 Requisitos

Os requisitos foram divididos primeiramente em funcionais e não-funcionais. Além disso, existem outras divisões relacionadas a hardware, software, estética e módulos do sistema. Para cada requisito foi dado um peso, conforme avaliação da relevância deste requisito para este projeto, de modo que no topo da árvore haja uma soma equivalente a 1.

#### 2.1.3.1 Requisitos Funcionais

##### 2.1.3.1.1 Requisitos de hardware

É importante observar que estes dividem-se entre a lamparina, que serve de ponto de acesso, e o módulo de celular, com pesos iguais entre os dois, já que a comunicação é bilateral e ambos têm importância equivalente. Há requisitos em comum entre esses subsistemas:

##### 2.1.3.1.2 Requisitos de software

Dividem-se entre a lamparina, o módulo de celular e o aplicativo, utilizado para receber e enviar dados para este módulo. Existem requisitos que são complementares às funções de hardware, porém correspondentes às soluções de software adotadas, como a implementação da norma IEEE 802.15.7 e a comunicação full-duplex um a um. Além destes, existem os requisitos de software comuns tanto à lamparina, ou Access Point Li-Fi, como ao módulo de telefone:

#### 2.1.3.2 Requisitos Não-Funcionais

##### 2.1.3.2.1 Conclusão

## 2.2 Software

Explica as escolhas feitas no aspecto do software do projeto.





## 3 Execução

5.1 Controle abre-fecha e rotação da garra Primeiramente, foi feito o estudo da garra e de seu controle, uma vez que seu controle é totalmente independente da posição do braço. Para os testes, a fim de prova de conceito, foram usadas entradas (abre-fecha, graus de rotação) geradas pelo próprio computador.

Figura 4: Garra do braço robótico 5.2 Modelagem de movimentação do braço Foi feita a modelagem de todos os movimentos possíveis para o braço robótico. Esses resultados serão posteriormente usados como saída para o treinamento da ANFIS. 5.2.1 Modelagem da movimentação do braço para os motores 1 e 2, correspondentes ao ombro e ao cotovelo do braço Foi feita inicialmente a modelagem para dois motores (ombro e cotovelo), adaptando-se o modelo da referência [7] ao braço a ser utilizado.

Figura 5: Coordenadas X-Y para todos os valores de  $\theta_1$  e  $\theta_2$  5.2.2 Modelagem da movimentação do braço para os motores 1, 2 e 3 correspondentes ao ombro, cotovelo e rotação do ombro do braço O modelo foi ampliado para incluir o terceiro dos 4 motores de movimento do braço.

Figura 6: Coordenadas X-Y-Z para todos os valores de  $\theta_1$ ,  $\theta_2$  e  $\theta_3$  5.2.3 Modelagem da movimentação do braço para os motores 1, 2, 3 e 4 correspondentes ao ombro, cotovelo, rotação do ombro e pulso do braço

Inclusão do quarto e último motor referente ao braço, completando o modelo a ser utilizado.

Figura 7: Coordenadas X-Y-Z para todos os valores de  $\theta_1$ ,  $\theta_2$ ,  $\theta_3$  e  $\theta_4$

5.3 Modelagens do servo motor Para a modelagem do servo, foram feitas modificações no servo, que permitiram a leitura da posição em tempo real do motor. Dessa forma, é possível a comparação da saída do sistema em relação à entrada, possibilitando uma estimativa de uma função de transferência simplificada, assim como a constatação de alguma não linearidade do sistema.

Figura 8: Servo motor após modificação A partir disso, foi feita a aquisição da resposta ao degrau do mesmo sendo utilizado um Arduino MEGA 2560 como interface.

Figura 9: Montagem Experimental para determinação do modelo do servo motor

Figura 10: Saída obtida à partir da aplicação de degrau como entrada Os dados obtidos foram então inseridos no programa Matlab, em que através da ferramenta System Identification foi determinado o modelo do servo motor.

Figura 11: Utilização da ferramenta System Identification do Matlab para determinação de modelo

Figura 12: Modelo Linearizado Obtido Figura 13: Diagrama de Polos e Zeros do Modelo Obtido

Figura 14: Resposta ao degrau do Modelo Obtido

5.4 Modelagem do controlador

Para a modelagem do controlador, é proposta uma abordagem usando-se a planta real, a fim de capturar todas as não linearidades da planta, sendo o modelo linearizado usado para estimativa inicial dos parâmetros dos controladores PID e PIV puros. Devido ao alto custo computacional e a necessidade de modificação do hardware, esta abordagem foi interrompida.

Figura 15: Controlador PID

Figura 16: Controlador PIV

### 5.5 Aquisição de imagens da camera

Figura 17: Imagem das cameras estereoscópicas

A aquisição das imagens foi feita usando um computador Raspberry Pi rodando o sistema operacional Raspbian Jessie. Para isso, foram testados os programas motion, gstreamer e Mjpg-Streamer. A seguir, é feita uma comparação dos programas: motion: baixa complexidade de configuração, taxa de captura baixa, alto consumo de capacidade de processamento. Foi logo descartado Gstreamer: alta complexidade para configuração inicial, taxa de captura moderada, alto consumo de capacidade de processamento, não amigável ao uso de duas câmeras. Mjpg-Streamer: Configuração inicial de dificuldade moderada, taxa de captura adequada quando configurado para modo de baixa resolução. Após a escolha do programa, foi feito o estudo para a captura em tempo real de duas câmeras ao mesmo tempo. Para isso, foi primeiro feito o teste para duas instâncias do programa rodando ao mesmo tempo, após ajuste de taxa de captura e resolução provou-se que o programa mantinha-se estável o suficiente para aplicação. No anexo A estão disponíveis os códigos utilizados para a inicialização do programa MJPG-Streamer e o código da página HTML usada para juntar as imagens das duas câmeras

### 5.6 Montagem do sistema de motores e controle dos motores

Após a configuração das câmeras, foi feita a montagem de sua base com motores, que pode ser vista abaixo:

Figura 18: Sistema de câmeras com seus componentes

Após a montagem, foi feita a configuração de duas placas Particle Photon que foram utilizadas como transmissor e receptor, devido ao seu tamanho e peso reduzidos e ao fato de possuir uma antena WiFi já embutida. Para a captura dos movimentos da cabeça do usuário, decidiu-se acoplar um giroscópio externo aos óculos de realidade virtual, de modo que o projeto não

dependesse de formas de captura de movimentos específicas para cada modelo de óculos. Para o transmissor, foram realizadas duas versões de código. Inicialmente, foi escrito um código usando-se a saída do giroscópio e realizando as contas usando-se o microcontrolador conforme [23]. Este código era altamente dependente da posição inicial do giroscópio, que era usada para uma rotina de calibragem inicial. (Anexo B)

A segunda versão do código foi realizada usando-se bibliotecas específicas ("MPU6050/I2Cdev.h" "MPU6050/MPU6050\_6Axis\_MotionApps20.h") traduzidas de arduino, que fazem uso de algoritmos internos ao giroscópio para a calibragem, tornando-a mais rápida e precisa (Anexo B)

Para o receptor, foi feita uma única versão, responsável por receber os ângulos de rotação e transmiti-los aos dois motores na base da câmera (Anexo B). A transmissão dos dados atualmente se dá via protocolo UDP em uma rede local.

Figura 19: Óculos de realidade virtual com destaque para o giroscópio externo

5.7 Captura de movimentos do braço do usuário Foi utilizado inicialmente a linguagem Processing (baseada em Java), usando-se da biblioteca OpenNI para a captura dos movimentos. No entanto, devido a baixa precisão da captura usando a biblioteca.,

Figura 20: Captura de ângulos usando Processing A programação foi realizada em Visual C# usando-se como base o exemplo de captura e visualização de esqueleto do Microsoft Kinect 1.8 fornecido pela Microsoft, ao qual foram adicionadas funcionalidades necessárias ao projeto. No estado atual, o programa pode além de exibir o esqueleto com as juntas do usuário, capturar os ângulos do braço do usuário e exibi-los como números, para que seja feito o teste de coerência dos dados a serem enviados. Os códigos usados atualmente encontram-se no Anexo C.

Figura 21: Aquisição de Ângulos dos braços

5.8 Controle dos servomotores via conexão serial Foi feita a integração dos servomotores com o programa responsável pela captura dos movimentos do usuário via conexão serial USB. Pretendia-se para esta etapa o início do teste em rede local, mas devido a atrasos no envio do shield Wifi ESP8266 e a necessidade de reposição de peças do braço, houve atrasos no projeto. Como próximos passos, deseja-se implementar a possibilidade de uso do braço usando tecnologia WiFi e adaptar o código ao Kinect 2.0.

Figura 22: Teste do motor da base do braço



## 4 Conclusão

Conclusão pendente para o fim do projeto.



# Referências

ALEOTTI, J.; SKOGLUND, A.; DUCKETT, T. Gesture controlled mobile robotic arm using accelerometer. *International Journal of Innovative Research in Science, Engineering and Technology*, v. 4. Disponível em: <<http://aass.oru.se/~asd/Aleotti04IMG.pdf>>. Citado na página 21.

BRUMSON, B. *Robotics in Security and Military Applications*. Robotic Industries Associaton, 2011. Disponível em: <[http://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Robotics-in-Security-and-Military-Applications/content\\_id/3112](http://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Robotics-in-Security-and-Military-Applications/content_id/3112)>. Acesso em: 6 mar 2016. Citado na página 21.

CHANIOTAKIS; CORY. *Operational Amplifier Circuits Comparators and Positive Feedback*. 2006. Disponível em: <[https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-071j-introduction-to-electronics-signals-and-measurement-spring-2006/lecture-notes/24\\_op\\_amps3.pdf](https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-071j-introduction-to-electronics-signals-and-measurement-spring-2006/lecture-notes/24_op_amps3.pdf)>. Acesso em: 16 nov 2016. Nenhuma citação no texto.

CSA. *Uses for Robotic Arm Technologyi*. Canadian Space Agency, 2011. Disponível em: <<http://www.asc-csa.gc.ca/eng/canadarm/robotic.asp>>. Acesso em: 22 abr 2016. Citado na página 21.

DU, G. et al. Markerless kinect-based hand tracking for robot teleoperation. *IEEE*, v. 33, 2012. Disponível em: <[http://cdn.intechopen.com/pdfs/37895/InTech-Markerless\\_kinect\\_based\\_hand\\_tracking\\_for\\_robot\\_teleoperation.pdf](http://cdn.intechopen.com/pdfs/37895/InTech-Markerless_kinect_based_hand_tracking_for_robot_teleoperation.pdf)>. Acesso em: 6 mar 2016. Citado na página 21.

GEISEL, W. A. Tutorial on reed-solomon error correction coding. NASA, 1990. Disponível em: <<http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19900019023.pdf>>. Acesso em: 19 mai 2016. Nenhuma citação no texto.

GUPTA, S. Research on li-fi technology& comparison of li-fi/wi-fi. 2015. Disponível em: <[http://www.ijarcse.com/docs/papers/Volume\\_5/6\\_June2015/V5I6-0175.pdf](http://www.ijarcse.com/docs/papers/Volume_5/6_June2015/V5I6-0175.pdf)>. Acesso em: 6 mar 2016. Nenhuma citação no texto.

ICOR. *CALIBER MK4 LARGE EOD ROBOT Datasheet*. [S.l.], 2016. Disponível em: <<https://icor-b23.kxcdn.com/wp-content/uploads/2016/09/CALIBER%C2%AE-MK4-V2016.08.25.pdf>>. Acesso em: 16 nov 2016. Citado na página 21.

IEEE. *802.15.7-2011 - IEEE Standard for Local and Metropolitan Area Networks-Part 15.7: Short-Range Wireless Optical Communication Using Visible Light*. [S.l.], 2011. Disponível em: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6016195>>. Acesso em: 6 mar 2016. Nenhuma citação no texto.

INSTRUMENTS, T. *Precision, High Speed Transimpedance Amplifier*. [S.l.], 2007. Disponível em: <<http://www.ti.com/lit/ds/symlink/opa380.pdf>>. Acesso em: 16 nov 2016. Nenhuma citação no texto.

MITSANTISUK, C.; KATSURA, S.; OHISHI, K. Sensorless interaction force control based on b-spline function for human-robot systems. *SICE Journal of*

*Control, Measurement, and System Integration*, v. 1. Disponível em: <[https://www.jstage.jst.go.jp/article/jcmsi/1/6/1\\_6\\_452/\\_pdf](https://www.jstage.jst.go.jp/article/jcmsi/1/6/1_6_452/_pdf)>. Citado na página 21.

SEDRA/SMITH. *Microelectronics Circuits*. 7. ed. [S.l.]: Oxford, 2014. ISBN 0199339139. Nenhuma citação no texto.

VISHAY. *IRLZ14, SiHLZ14 Datasheet*. [S.l.], 2011. Disponível em: <<http://www.vishay.com/docs/91325/sihlz14.pdf>>. Acesso em: 16 nov 2016. Nenhuma citação no texto.

WANG, Y. et al. Robot teleoperation using a vision-based manipulation method. *IEEE*, v. 33, 2005. Disponível em: <<http://ieeexplore.ieee.org/abstract/document/5685167/>>. Acesso em: 6 mar 2016. Citado na página 21.