

# Protótipo de braço robótico capaz de capturar objetos fazendo uso de processamento de imagens e redes neurais

ARAUJO, Pedro Filipe C<sup>1</sup>  
SOUZA, Luan Vieira de<sup>2</sup>

<sup>1</sup>Acadêmicos do 6º período do Curso de Ciência da Computação, UNIFENAS, Alfenas.

## RESUMO

Este artigo apresenta o desenvolvimento de um braço robótico cujos movimentos são controlados por um sistema, e os resultados parciais até aqui obtidos da implementação de uma rede neural para que este fosse capaz de si mover de maneira autônoma. Seus movimentos emulam os movimentos de um braço industrial na fabricação ou na captura de peças, objetos, etc. A comunicação entre o braço e o sistema foi feito por meio de um dispositivo denominado nodeMCU, uma placa que possui um módulo WiFi integrado e foi configurada para trabalhar como um servidor que recebe requisições e às trata passando aos motores do braço. Para os testes com o reconhecimento automático de objetos, foi utilizado uma rede neural convulacional com arquitetura SSD(Single Shot Multibox Detector) para realizar o processamento das imagens capturadas pela câmera. Os resultados ainda não estão completos pois o projeto ainda se encontra em desenvolvimento.

## ABSTRACT

This article presents the development of a robotic arm whose movements are controlled by a system, and the partial results obtained so far from the implementation of a neural network so that it could be able to move autonomously. Their movements emulate the movements of an industrial arm in the manufacture or capture of pieces, objects, etc. The communication between the arm and the system was done through a device called nodeMCU, a board that has an integrated WiFi module and was configured to work as a server that receives requests and treats them passing to the arm motors. For the tests with the automatic recognition of objects, a neural network was used convulacional with architecture SSD (Single Shot Multibox Detector) to realize the processing of the images captured by the camera. The results are not yet complete because the project is still under development.

## PALAVRAS-CHAVE

Braço robótico, braço inteligente, nodemcu, processamento de imagens, redes convolucionais.

## KEYWORDS

Robotic arm, intelligent arm, nodemcu, image processing, convulational networks.

## 1 INTRODUÇÃO

Nestes últimos anos, com o avanço tecnológico, o crescimento de profissionais especializados e redução de custos de componentes eletrônicos [1], o campo da robótica vem se popularizando e se difundindo cada vez mais nos países em desenvolvimento, principalmente nas indústrias automobilísticas. Segundo a IFR (International Federation of Robotics) [2], em 2011, cerca de 60.000 unidades de braços robóticos foram vendidas no mundo e, em 2013, 70.000. Como reflexo dessa popularização, muitos grupos de pesquisadores e interessados no assunto começaram a investir nessa área através do uso dos robôs como instrumento de auxílio para suas atividades bem como para o ensino técnico focado no seu funcionamento.

Este trabalho está organizado de acordo com a seguinte estrutura:

- Seção 2 apresenta os métodos empregados para a integração dos componentes e a arquitetura da rede neural utilizada;
- Seção 3 apresenta alguns resultados após o processamento da imagem obtida por uma câmera;
- Seção 5 apresenta as conclusões geradas pela observação dos resultados e implementação do sistema.

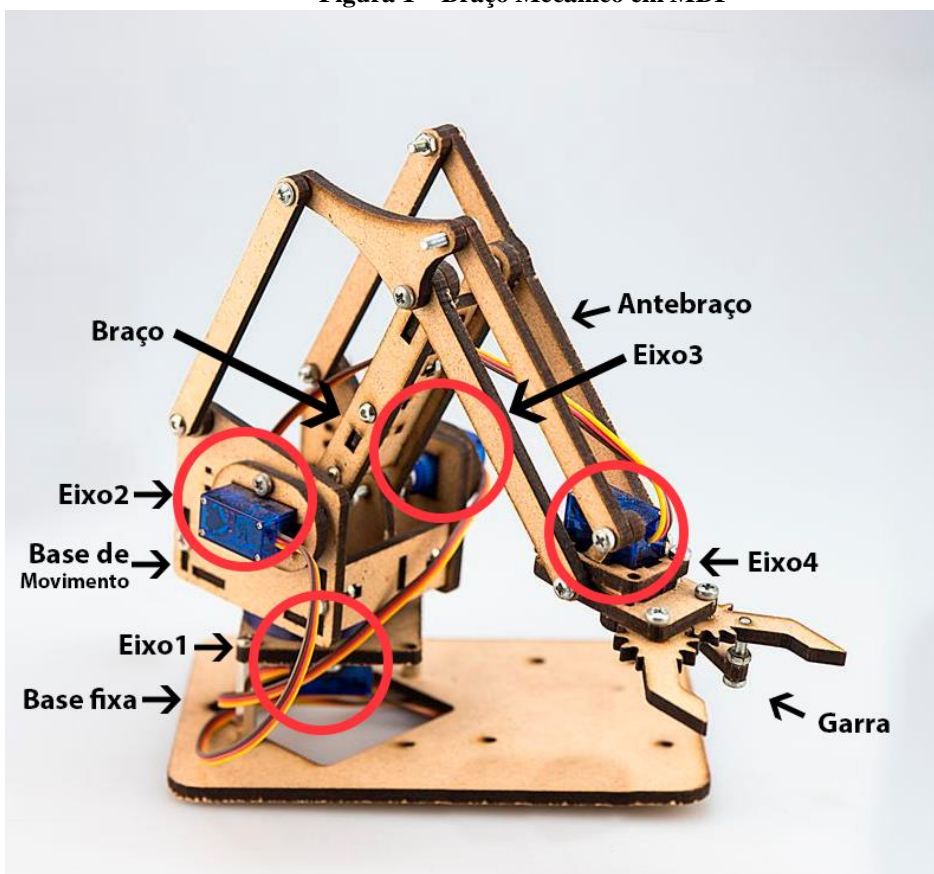
## 2 METODOLOGIA

### 2.1 Parte estrutural

Foi utilizado um braço composto de MDF que realiza diversos movimentos possuindo 4 graus de liberdade (DOF).

Neste tópico será demonstrado a função de cada movimento que o braço pode realizar em 180° na horizontal e na vertical.

**Figura 1 – Braço Mecânico em MDF**



As principais partes do braço mecânico estão indicadas na Figura 1, suas funções serão escritas a seguir:

- Base fixa – Sua Função é fixar o eixo 1 e criar estabilidade para os movimentos do braço.
- Eixo 1 - Tem a função de realizar os movimentos na horizontal da base de movimento.
- Eixo 2 - Tem a função de realizar os movimentos do braço na vertical;
- Eixo 3 - Tem a função de realizar os movimentos do antebraço na vertical;
- Eixo 4 - Tem a função de abrir e fechar a garra.

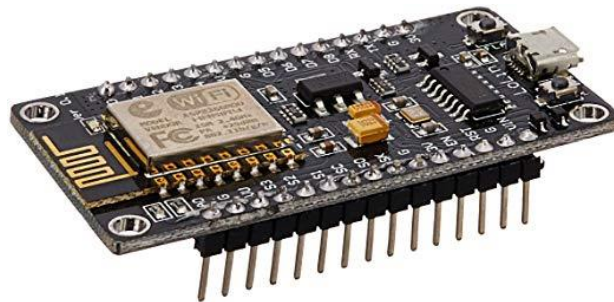
### 2.2 Parte elétrica

Foram utilizados para a parte elétrica do braço 4 micro servos motores acoplados no braço mecânico para realizar os movimentos .



**Figura 2 – Servo Motor**

NodeMCU foi a placa escolhida, pois ela já possui o módulo WiFi integrado e permite que seja configurada para que trabalhe como um meio de comunicação (servidor) entre o computador e os servo motores, recebendo requisições e as tratando enviando de volta para os servos ou para o computador



**Figura 3 – Nodemcu**

Uma fonte com tensão de 5volts e corrente de 2amperes foi necessária para alimentar os motores, já que o nodeMCU por si só, não é capaz de suprir esta demanda para o bom funcionamento dos motores.



**Figura 4 - Fonte de computador**

Uma câmera foi utilizada para efetuar a verificação do meio em tempo real, sendo este o componente que se comunica diretamente com a rede neural para que o processamento e reconhecimento de objetos seja possível.



Figura 5 – Câmera

### 2.3 Web Service no NodeMCU

A base para que tudo fosse realizado posteriormente foi justamente o trabalho feito no NodeMCU. Como ele seria o meio de comunicação entre os servo motores do braço e o sistema desenvolvido, havia a necessidade de uma atenção especial a ele para que esta comunicação fosse simplificada. Pensando nisso foi levantado nele um Web Service, com rotas para requisições GET e POST, fazendo com que qualquer sistema fosse capaz de enviar requisições a ele e recebesse, tudo no formato JSON.

192.168.4.1

---

**Bem vindo ao REST Server do Braco Mecanico**

**GET /servo**  
PARAMETROS: num  
EXEMPLO: /servo?num=1  
RETORNO: { 'num': 1, 'gpio': 4, 'posicao' : 0 }

**POST /servo/config**  
PARAMETROS: num, gpio  
EXEMPLO: { 'num': 1, 'gpio':4}

**POST /servo/posicao**  
PARAMETROS: num, posicao  
EXEMPLO: { 'num': 1, 'posicao':180}

Figura 6 – GET/POST

A seguir a descrição de cada rota reconhecida pelo NodeMCU e o seu funcionamento:

1. GET /servo

Por meio deste GET e a passagem do número do servo que deseja capturar as informações, é possível saber o posicionamento em graus de onde o servo motor está e em qual gpio(porta) ele está conectada no node;

## 2. POST /servo/config

Foi implementada para tornar a aplicação flexível, pois nada dentro do nodeMCU é fixa. Pela passagem de parâmetros deste POST, é possível realizar a configuração de cada servo sem a necessidade de realiza-la via código tornando-a uma aplicação mais escalável.

## 3. POST /servo/posição

Por meio deste é possível realizar a movimentação dos motores. Basta passar o número do servo motor a ser controlado e o posicionamento do mesmo para que ele execute o comando.

## 2.4 Implementação do sistema

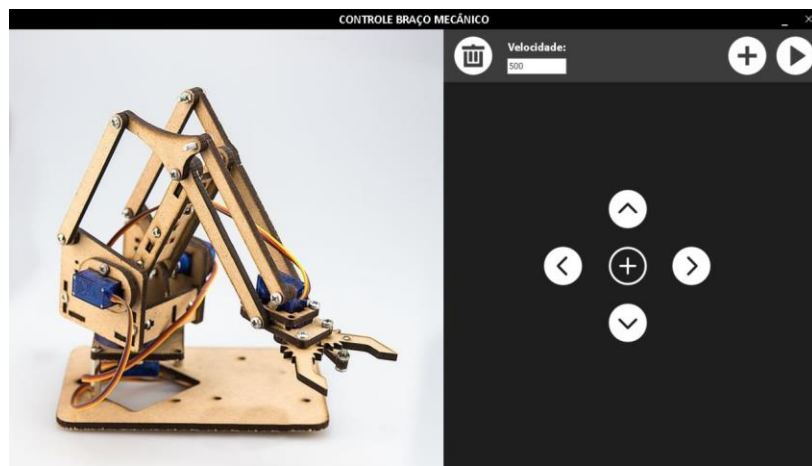


Figura 7 – Software

O software para o controle do braço foi desenvolvido em C#. Seu funcionamento é bem simples, bastando selecionar o motor pela imagem e realizar a sua movimentação com as setas. Além disso, foi implementado uma funcionalidade onde é possível ao usuário salvar uma série de movimentos para o braço ficar reproduzindo em loop, além de poder também definir o tempo de intervalo entre um movimento e outro para movimentos mais rápidos ou mais lentos.

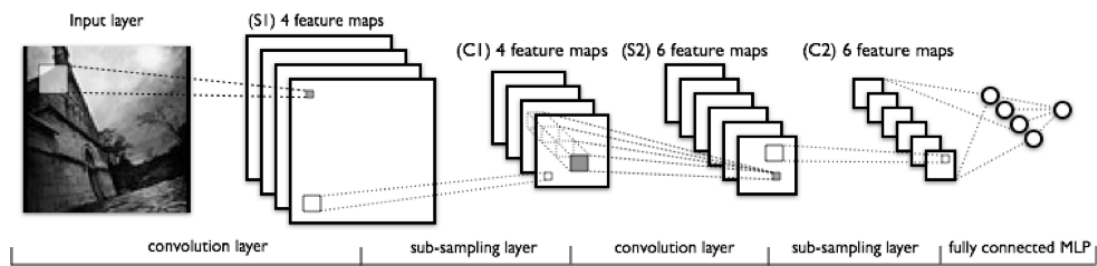
## 2.5 Redes neurais convolucionais

É muito fácil para o nós humanos realizarmos o reconhecimento de objetos, mas os computadores tem uma certa dificuldade com esta mesma tarefa. Quando olhamos para algo como um lápis, um carro ou uma pessoa, geralmente não precisamos estudá-lo conscientemente antes de podermos dizer o que é. No entanto, para os computadores, identificar qualquer coisa representa um problema muito difícil, e a maneira de resolver este problema seria através da utilização de Redes Neurais Artificiais(ANN).

Na teoria poderíamos fazer uso das redes neurais convencionais para analisar imagens, mas na prática, elas acabam gerando uma carga computacional que as tornam inviáveis. Por exemplo, uma rede neural convencional tentando processar uma imagem pequena (30x30 pixels), precisaria de 0,5 milhões de parâmetros e 900 entradas. Foi desta necessidade de fazer com que o processamento se tornasse computacionalmente gerenciável que surgiram as redes neurais convolucionais.

Elas detectam pontos de interesse na imagem por meio da análise de seus pixels, e a partir daí fazem com que cada camada trabalhe em um ponto, diferente das redes convencionais onde cada camada tem a tarefa de processar a imagem

toda. Isto faz com que o processamento se torne muito mais rápido, já que ela irá processar somente os pontos de interesse e não a imagem toda.



**Figura 8 - Rede convolucional**

Da esquerda para a direita na imagem acima, podemos observar:

- Input layer - A imagem de entrada real que é digitalizada.
- O filtro que passa sobre a imagem de entrada é o retângulo da luz.
- S1 - Os mapas de ativação estão dispostos em uma pilha em cima um do outro, um para cada filtro utilizado.
- O retângulo maior é 1 trecho para ser amostrado.
- C1 - Os mapas de ativação condensados através de amostragem descendente.
- S2 - Um novo grupo de mapas de ativação gerado passando os filtros sobre a pilha que é amostrada em primeiro lugar.
- C2- A segunda amostra - que condensa o segundo grupo de mapas de ativação.
- Uma camada totalmente conectada que designa saída com 1 rótulo por nó.

## 2.6 A arquitetura da rede proposta

Com base nas pesquisas realizadas, foi obtida a conclusão de que a melhor arquitetura a ser utilizada para atender ao problema seria a SSD (Single Shot Multibox Detector). Ela é one-shot, ou seja, uma vez estando treinada, ela é capaz de chegar a um resultado em uma execução só, o que a torna indicada para casos onde há a necessidade do reconhecimento em tempo real de objetos, tendo uma precisão de 85% à 90%. Ela tem dificuldades quando o objeto está muito longe, mas para a identificação dos objetos do braço robótico ela irá atender muito bem.



**Figura 9 – Modelo de Identificação IA**

### 3 RESULTADOS E DISCUSSÃO

- Resultados obtidos inicialmente foram satisfatórios. A parte de identificação ainda não foi feita por meio de redes neurais, mas houve um processamento dos pixels da imagem.
- Eram dois parâmetros que o algoritmo deveria analisar: o centro, e o objeto. Quando este objeto não estivesse no centro e o sensor de presença (regulado para reconhecer a presença somente quando o objeto estivesse bem próximo) o algoritmo deveria enviar sinais para o motor fazendo com que o objeto ficasse ao centro e o sensor de presença o identificasse.
- Ele só fecharia a garra quando os dois parâmetros fossem verdadeiros.
- A precisão de acerto foi de 45%, nos testes efetuados, pois ainda não foi implementado a rede para que ele realizasse a classificação dos objetos e capturasse o correto.
- Nos testes, quando mais de 3 objetos foram colocados no campo de visão, ele não conseguiu decidir em qual objeto ir.

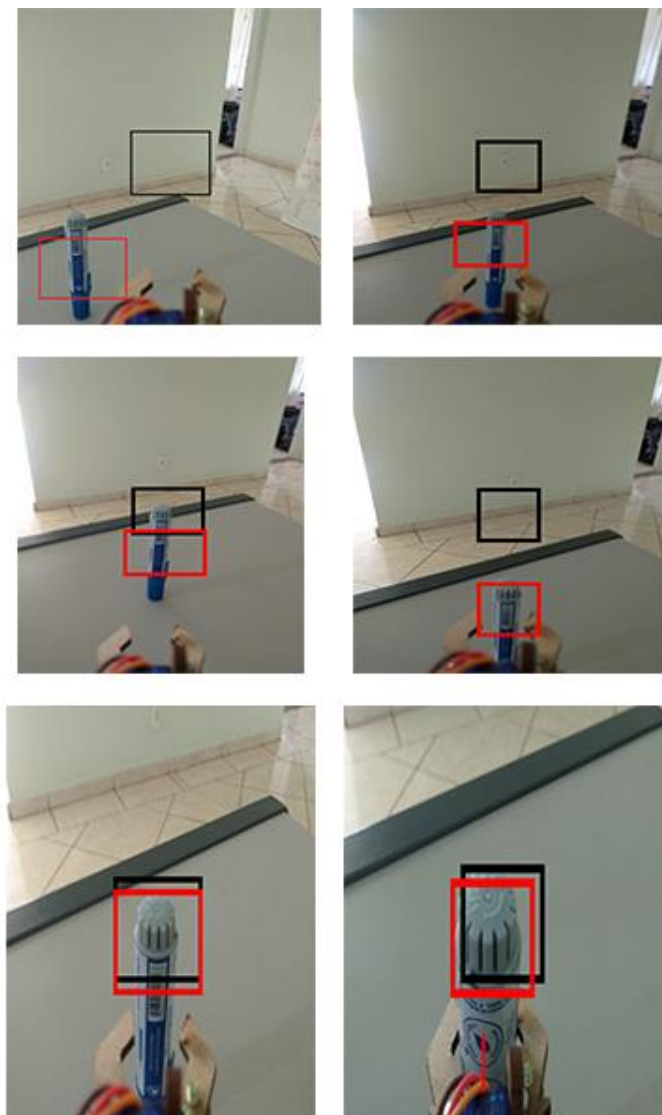


Figura 10 – Testes processamento de imagens

#### 4 CONCLUSÃO

Utilizando o mínimo de componentes foi demonstrado o potencial desta aplicação. Um braço mecânico pode ser muito útil em áreas onde há riscos ou que necessitem de precisão em suas operações, podendo ser manuseadas remotamente a longas distancias, ou programadas para executar determinadas rotinas repetitivas sem perder a eficiência.

Propõe-se como opção para estudos futuros, a utilização do protótipo do braço robotico para a simulação em produção de uma empresa onde há a necessidade de uma determinada precisão. Estudos poderão ainda utilizar processamento de imagens e Inteligência artificial para distinguir objetos.



**Figura 11 – Teste de movimentos repetidos.**

#### REFERÊNCIAS

- [1] Malone, T. W; Laubacher, R; Johns, T. The big idea: The age of hyperspecialization. Disponível em: <<https://hbr.org/2011/07/the-big-idea-theage-of-hyperspecialization/ar/1>>. Acesso em 25 de fevereiro de 2015.
- [2] IFR. International Federation of Robotics. Disponível em: <[ifr.org](http://ifr.org)>. Acesso em 25 de Fevereiro de 2015.