

# Sistema de Controle de Braço Robótico para Reprodução de Movimentos via Reconhecimento de Voz e Imagem

Douglas Fidelis Lopes, Daniel Pereira Nascimento, Fabricio do Carmo Santana, Nicolau Harabura, Rafael Alexandre das Graças, Leonardo Cassiolato Silva, Marcel Stefan Wagner

Universidade Anhembi Morumbi (UAM)  
Departamento de Engenharia da Computação

**Resumo** — A Robótica é uma ferramenta que está ligada grandemente à revolução da nossa sociedade e nos meios de produção. Ao usá-la da maneira correta podemos melhorar a eficiência de trabalhos realizados, assim como preservar a saúde dos trabalhadores. Neste projeto de TCC foi desenvolvido um braço robótico que replica os movimentos de uma mão humana, utilizando o reconhecimento de imagem com uma câmera para captar o movimento e servo motores programáveis instalados no braço que replicam os movimentos captados. O braço foi impresso em 3D, baseado no braço InMoov, com o objetivo de encontrar maneiras alternativas de realizar atividades manuais que possam apresentar risco ao serem executadas diretamente por um humano.

**Palavras-Chave** — braço robótico; reconhecimento de imagem; OpenCV; servo motor; impressão 3D.

## I. INTRODUÇÃO

A robótica é uma área abrangente da tecnologia, sendo assim possui aplicação em diversos setores como: na produção industrial, com braços robóticos em linhas de produção, na medicina auxiliando em cirurgias e até em atividades domésticas. Segundo informações da revista Cipa, “Máquinas e equipamentos provocaram 528.473 acidentes de trabalho, tendo como consequência 2.058 mortes acidentárias notificadas e 25.790 amputações ou enucleações no Brasil entre 2012 e 2018” que são valores que, de certa forma, impressionam, pois não se imagina tantos acidentes e muito menos mortes dessa forma.

Seguindo essa linha de raciocínio este projeto visou um desenvolvimento exploratório ao buscar formas de auxílio em atividades em geral que possam ser realizadas por um braço robótico remotamente ao invés de um braço humano, visando minimizar riscos que existem ao manusear diretamente algum objeto.

O projeto é um sistema de controle de braço robótico impresso em 3D, baseado no projeto InMoov [1], utilizando a linguagem de programação Python com a biblioteca CVZone para reconhecimento de vídeo e a placa Arduino UNO [2] para recebimento dessas informações via cabo USB e execução de códigos programados. No Arduino também é realizada a conexão com o módulo de reconhecimento de voz Voice Recognition V3[3], para registro de comandos de voz.

Após receber essas informações, a placa Arduino envia instruções de rotação de até 180° para os servos motores, que durante este movimento, puxa cordas ligadas a cada dedo da mão robótica, executando o movimento especificado.

## II. PROBLEMA DE PESQUISA E OBJETIVOS

A impressão do braço foi o principal fator que impactou o desenvolvimento do projeto, pois requer um material de alto custo e uma quantidade considerável de tempo para sua conclusão, causando um início tardio do desenvolvimento do projeto.

No planejamento inicial, existiu a possibilidade de expandirmos o projeto incluindo outras partes robóticas ao “corpo”, todavia devido às dificuldades enfrentadas não foi possível desenvolver além do que está descrito neste documento.

Com exceção destes, não foram encontrados enormes problemas previamente ao desenvolvimento. Apesar de não ter sido possível expandir o projeto, a meta estabelecida como objetivo foi alcançada: buscar alternativas para auxiliar seres humanos na realização de atividades em diversos campos, com uso de reconhecimento de movimento através da captação de imagem e reconhecimento de voz e áudio para reprodução de movimentos de braço robótico, trazendo como consequência uma maior segurança para os usuários, não expondo-os diretamente aos riscos que podem existir ao executar determinada atividade.

## III. PESQUISA TEÓRICA

Em janeiro de 2012, o escultor e designer francês, Gaël LANGEVIN desenvolveu um projeto pessoal que recebeu o nome de InMoov. Este foi o primeiro projeto de prótese de mão *Open Source* e acabou por se desenvolver à um robô humanoide completo. Utilizando os *templates* 3D criados e disponibilizados pelo projeto InMoov para parte tangível do nosso projeto: a impressão do braço robótico, com adição de reconhecimento de imagem e de voz.

Existem projetos que foram realizados com esta mesma proposta, porém utilizam meios diferentes para obter o mesmo resultado, como a utilização de sensores em uma luva para identificação dos movimentos realizados por cada dedo e em seguida transmissão desses dados para uma placa Arduino para replicação no braço robótico. Outra solução, que neste caso é a solução utilizada neste projeto, é a reprodução por reconhecimento de imagem e áudio.

Esse método requer uso de software específico: em um ambiente de desenvolvimento (IDE), neste caso *PyCharm* [4], utilizando a linguagem Python e os módulos *CVZone* [5] para que seja feito o reconhecimento do movimento a ser executado

e posteriormente o *SerialDevice* para envio dessas informações visando execução via placa Arduino. O Arduino por sua vez é programado com linguagem C. Com a utilização do módulo de reconhecimento de voz *Voice Recognition V3 (VR3)*.

A placa V3 tem capacidade para armazenar até 80 comandos de voz cada um com duração de 1500 milissegundos. Este módulo compara as vozes com um registro de comandos de voz gravados anteriormente. É possível gravar o comando em qualquer idioma ou até mesmo qualquer som e usá-lo como um comando. Portanto, inicialmente é preciso treinar o módulo para posteriormente permitir que reconheça os comandos.

Assim se descreve o fluxo do sistema: Inicia-se com o reconhecimento de movimento da mão humana via vídeo em uma IDE, em seguida envia-se as informações para a placa Arduino e por fim o envio dos dados para execução de rotação dos servos motores. O fluxo alternativo se dá a partir do reconhecimento de voz, iniciando-se assim diretamente com o módulo V3 para reconhecimento juntamente ao Arduino e em seguida envio da instrução para reproduzir o movimento especificado com os servos motores.

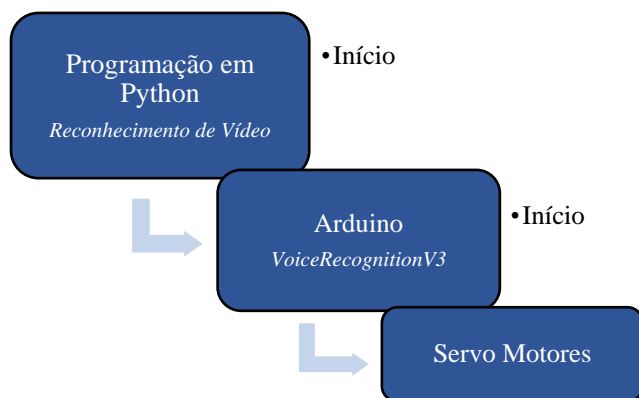


Fig. 1 - Fluxograma do Sistema.  
Fonte: Autores, 2022.

#### IV. METODOLOGIA

Para o reconhecimento de vídeo, na linguagem de programação Python, foi utilizado bibliotecas que possuem recursos já desenvolvidos para reconhecimento de imagem (*CvZone* e *OpenCV*).

Através da função '*findHands*' (cuja tradução é '*EncontrarMão*') do *CvZone*, é detectado a mão no quadro e localizado os pontos chave desenhando os pontos de referência, conforme a Figura 2.

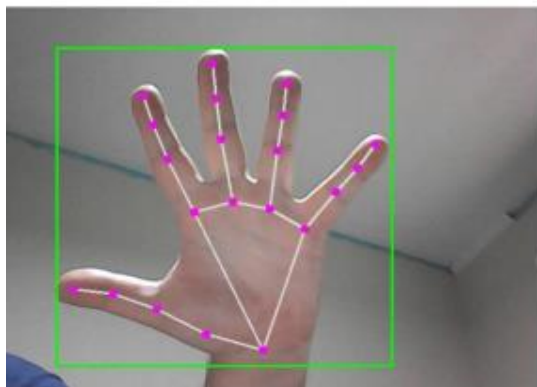


Fig. 2 - Reconhecendo a mão e marcando pontos de controle.  
Fonte: Autores, 2022.

Em seguida, o dado processado é enviado para a placa arduino, através do módulo *SerialDevice*.

Nos parâmetros da função é fornecido qual porta USB que o Arduino está conectado, o valor da frequência no fluxo de dados entre Arduino e Python. Assim fazendo a conexão corretamente para envio das informações. No Arduino IDE também devem ser estabelecidas a porta e a frequência de acordo para conexão bem-sucedida.

Na IDE do Arduino foi realizada a importação das bibliotecas do servo motor '*Servo.h*' e do módulo de voz V3 '*VoiceRecognitionV3.h*' que serão necessárias para execução.

Cada dedo da mão robótica está conectado a um servomotor (Mg995) por meio de uma linha de multifilamento. Ao girar, os servos puxam as linhas, que por sua vez, puxam o respectivo dedo, realizando por consequência o movimento de fechar a mão.

Os servos motores estarão conectados em uma fonte de alimentação. Cada servo motor executará um movimento específico dentro do conjunto da ação, cinco deles estarão encarregados de movimentar um dedo cada e um deles para a rotação do braço em seu próprio eixo.



Fig. 3 - Servomotor Mg995 Fonte: Eletrogate [6]. Acesso em ago. 2022.

Com o objetivo de passar a energia necessária para os servos motores sem trepidação no sistema, foi utilizado um regulador de tensão LM2596. Sendo possível assim a alimentação dos servos, evitando problemas e quedas do sistema.



Fig. 4 - Regulador de tensão ajustável LM2596 Fonte: Baú da Eletrônica [7], Acesso em ago. 2022

O módulo VR3 com microfone que registra comandos de voz, podendo assim dar comandos ao braço robótico oralmente com funções pré-programadas no Arduino. O funcionamento do Módulo dá-se por meio de comandos de voz de

aproximadamente 1500ms, uma ou duas palavras em média, com 7 comandos que podem ser utilizados por vez e com capacidade de armazenamento de até 80 comandos.



Fig. 5 - Módulo de Reconhecimento de Voz V3 com Microfone.  
Fonte: Elechouse [8]. Acesso em mar. 2022

O principal interlocutor entre todas as peças é o Arduino Uno. Se trata de uma placa baseada no microcontrolador ATmega328 (*datasheet*) e contém todos os componentes necessários para suportar o microcontrolador, simplesmente conectando a um computador pela porta USB ou alimentar com uma fonte ou com uma bateria.



Figura 6 - Arduino Uno. Fonte: Casa da Robótica [9]. Acesso em Abr. 2022

Após os passos anteriores, há inúmeras possibilidades que dependem somente da programação para que a mão seja funcional da forma que achar necessário.

## V. DESENVOLVIMENTO

O principal elemento do projeto é o braço robótico impresso em 3D, as peças devem ser manipuladas delicadamente, pois são frágeis e o tempo para concluir a impressão é longo. Com as peças do braço prontas, foram testados o Arduino, servo motores, reguladores de tensão e o módulo de voz antes da montagem final, para conferência do funcionamento de cada peça e esquematização de arranjo físico do sistema.

Algumas adaptações foram necessárias para armazenar todos os componentes do projeto, pois apenas os servos cabem dentro do braço robótico, logo, os outros componentes, como fontes de alimentação e placas, foram colocados ao redor do braço em uma caixa com laterais de acrílico. Além disso, o braço naturalmente deve ser sustentado pelo corpo, porém como neste projeto temos apenas o braço impresso, o próprio ficará fixado na caixa.

Os servos motores não estão ligados diretamente ao Arduino, pois há um limite de portas e pinos que se é possível encaixar. O sistema possui 6 servos motores e o módulo de

reconhecimento de voz V3, portanto o Arduino não possui uma quantidade de entradas e saídas suficiente para conectá-los e não possui capacidade para alimentação do sistema todo. Portanto, foi utilizado uma placa externa chamada *protoboard* para conectar todos os servos motores. A placa não interfere no sistema e deve ser considerada apenas como uma extensão dos pinos. Em vista disso, ficou esquematizado da seguinte forma: o Arduino que está conectado no computador via USB, recebe a informação de reconhecimento visual e envia a mesma para o servo motor que está conectado diretamente ao Arduino, porém, sua alimentação passa por um regulador de tensão ligada a *protoboard* (devido ao limite físico de portas do Arduino) e do regulador é transferido a tensão para o servo motor. São no total 6 reguladores de tensão, 1 para cada servo motor.

Todos os servos necessitam de uma alimentação entre 4,8V e 6V, no entanto, a placa Arduino não é capaz de prover toda essa quantidade para os servos. Portanto, para a alimentação de todos é utilizada uma fonte externa de 6V conectada à *protoboard*.

Para realizar as conexões entre os reguladores de tensão, *protoboard*, placa Arduino e servos, foram utilizados cabos chamados de *jumper*. Os *jumpers* são peças fundamentais com funções como: desviar, ligar ou desligar o fluxo elétrico, fazendo com que seja cumprida as configurações específicas do projeto.

O código em Python realiza todo o reconhecimento de vídeo e envia em tempo real para o Arduino que coordena toda a função física. Cada servo executa o movimento independentemente do outro, no entanto, a execução de somente um também é possível.

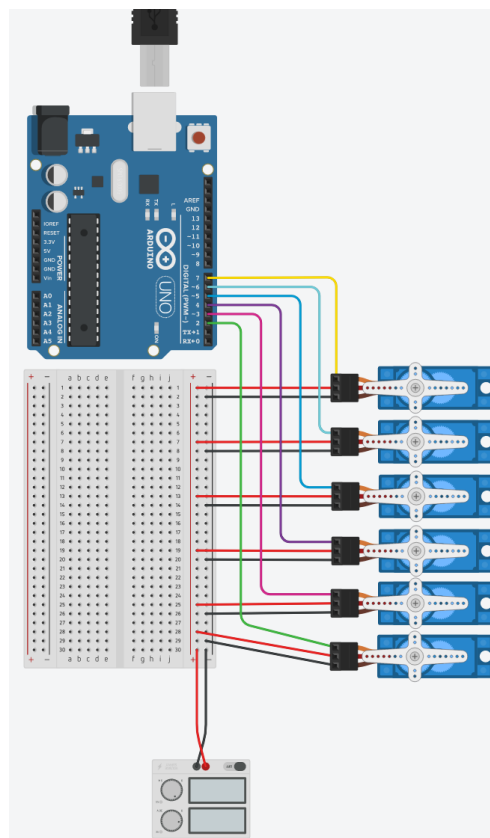


Fig. 7 - Simulação básica e parcial do sistema.  
Fonte: Autores, via TinkerCad [10], 2022.

Vale ressaltar que, conforme a Figura 7, foi utilizado o software *TinkerCad* para uma simulação do sistema, entretanto, não foi possível incluir nesta visualização o módulo de voz, a fonte de alimentação de 9V que alimenta a placa Arduino e por fim os reguladores de tensão que atuam entre a fonte de 6V e os servo motores.

Para a montagem foram impressas as peças: *Auticulaire3*, *CableHolderWristV5*, *RobCableBackV3*, *RobCableFrontV3*, *RobRingV3*, *RobServoBedV6*, *RotaWrist3V3*, *RotaWrist2V3*, *servo-pulleyX5*, *TensionerRightV1* e *WritGearsV5* do InMoov para armazenar os servos, pois são suportes que se encaixam dentro do antebraço.

Para o alinhamento das linhas que ligam os dedos aos servos, distribuímos as linhas para que não haja conflito entre elas. Para abrir e fechar a mão foi utilizada uma linha de pesca trançada multifilamento de 0,45mm, passada por dentro de cada peça ligada aos servos. Em cada dedo foram necessárias duas passagens de linhas, uma na parte traseira e outra na parte frontal interna de cada dedo. Via reconhecimento de voz ou imagem o servo gira 180° puxando as linhas e consequentemente a mão abrirá ou fechará os dedos. No pulso, 1 servo motor é designado para a rotação.

#### A. Configurações do Arduino

Começando pelo Arduino, é necessário instalar o programa do mesmo no computador em que irá conectar o Arduino (Obs: Foi usado a porta USB 3.0 para conectá-lo ao computador). Para alimentação da placa Arduino foi utilizado uma fonte de alimentação externa de 9V. Depois de instalar e executar o programa do Arduino foi necessário configurar no software a porta utilizada para fazer a troca de informações entre o computador e o Arduino. Para verificarmos qual a porta utilizada, foi usado o gerenciador de dispositivos da máquina e para identificar as portas “COM e LPT”, nela foi indicado a porta. (OBS: Para que essa opção fosse visível e seja possível programar o Arduino, é preciso que esteja conectado ao computador), depois da localização da porta foi iniciado o programa do Arduino. Assim que iniciado o programa foi selecionada a placa que está sendo utilizada, que neste caso é a Arduino Uno e foi configurada a porta.

Com os passos anteriores já concluídos e todas as peças conectadas, foi desenvolvido o código para a transmissão de dados para os servos motores. Seis servo motores foram utilizados para que haja movimento em todo o braço. Cinco servos com uma rotação de 180° estão responsáveis pelo movimento dos dedos e o sexto, com rotação de 90°, responsável pelo movimento de girar o pulso. Foi nomeado cada servo de acordo com a parte da mão que irá movimentar e após nomear atribuímos a taxa de transmissão de informações de 9600 bit/s. Esses foram os primeiros passos para o funcionamento do Arduino.

#### B. Fluxo via Reconhecimento de Vídeo

Uma das bibliotecas usadas no Python foi a *CVZone* na versão 1.1. “Biblioteca” neste caso é uma coleção de subprogramas utilizados no desenvolvimento de software. Bibliotecas contém código e dados auxiliares, que provém serviços a programas independentes, permitindo o compartilhamento e a alteração de código e dados. *CVZone* se trata de uma biblioteca que consegue identificar as partes do

corpo humano e as ações que foram realizadas em tempo real. É através desta biblioteca que o *software* consegue identificar os movimentos da mão, e com isso, pode passar essa informação para o Arduino. O motivo de fazer uso da versão 1.1 é devido as novas versões não possuírem a função “*findPosition*”. Portanto foi necessário a utilização de uma versão mais antiga. Outra biblioteca usada foi a *SerialDevice* na versão 0.2. ela serve para fazermos a conversação do Arduino com o Python e passar as informações utilizadas no processo. Servo é mais uma das bibliotecas, que serve para o Arduino poder se comunicar com os servos e passar as informações sobre a variação de rotação que devem fazer.

No sistema de controle de braço robótico foi utilizada a linguagem Python para executar a parte da programação, com uma biblioteca para reconhecimento de vídeo, captando o movimento das mãos e dos dedos. Ao identificá-los, vai ser atribuído o valor 0 ou 1, que representam as posições que os dedos podem assumir: fechado ou aberto. Esses valores serão armazenados em um vetor, onde cada índice representa um dedo, e o valor representa o movimento. As informações coletadas são transmitidas em tempo real para o Arduino, cuja conexão é feita via porta USB. O Arduino, ao receber esse vetor com valores 0 e 1, entende que 0 seria para fechar e 1 para abrir o dedo. Cada servo motor vai executar o movimento de abrir e fechar o dedo. O servo motor vai fazer o movimento de rotação de 180° graus, de acordo com o recebimento de informações do Arduino, que por sua vez, recebe informações do código em Python. Assim finalizando o fluxo de execução via reconhecimento de vídeo.

#### C. Fluxo via Reconhecimento de Voz

Além do Arduino ter as funções de movimento de cada motor foi implementado a biblioteca *VoiceRecognitionV3*, para tornar possível gravar comandos de voz via seu módulo físico. Assim, será possível associá-los a alguma função pré-definida e específica. Atribuindo esse comando gravado a uma função, quando reproduzir novamente esse comando, será identificado a função definida anteriormente. O comando só é reconhecido se for reproduzido pela mesma pessoa, ou seja, precisa primeiramente realizar o processo de treinamento no software do Arduino, gravando a voz do usuário para depois atribuí-la como entrada para utilização de funções pré-definidas. O módulo VR3 é uma placa física que possui um microfone conectado a si conecta-se diretamente à placa Arduino.

Desta forma, ao utilizar o reconhecimento de voz, o fluxo do sistema se inicia diretamente com o Arduino e o módulo VR3 e não necessita do código programado em Python.

#### D. Testes Finais



Fig. 8 - Braço Robótico.  
Fonte: Autores, 2022.



Para o reconhecimento da mão foi utilizada a IDE *PyCharm* com código em Python para enviar ao Arduino e assim replicar os movimentos.

Na Figura 9 é possível ver as marcações na mão e os pontos. Os pontos são exatamente nas articulações da mão e ponta dos dedos. É passado a informação do Arduino para o servo motor, assim movimentando o dedo específico. Com a demarcação da mão feita, definimos o reconhecimento se a mão está aberta ou se ela está fechada. No código é retornado o valor “1” ou “0” que representa que o dedo está esticado ou dobrado.

O valor retornado vem em forma de um vetor como o exemplo: “[1, 1, 1, 1, 1]” onde cada posição representa um dedo e o valor representa o estado.

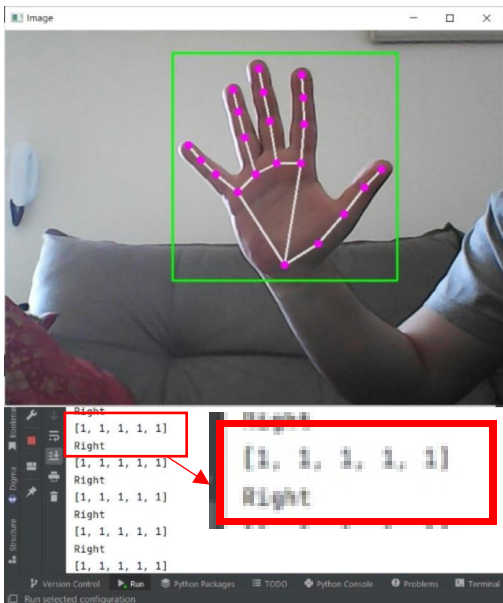


Fig. 9 - *PyCharm* reconhecendo a mão aberta e informando vetor com valores 1 para retorno.

Fonte: Autores, 2022.

No recorte em vermelho é destacado o vetor de valores 1 e a mão direita (*Right*) identificados, que é a saída do código de reconhecimento para o Arduino. Na Figura 10, é mostrado o resultado esperado do movimento no braço robótico.



Fig. 10 - Resposta da mão ao receber vetor de valores 1.

Fonte: Autores, 2022.

Quando é retornado “0” em todos os valores do vetor, é identificado que todos os dedos estão dobrados, por consequência, a mão está fechada. Após esse reconhecimento de aberta ou fechada, a mão já está funcional e replicando os movimentos de acordo com a imagem reconhecida pelo *Pycharm*.

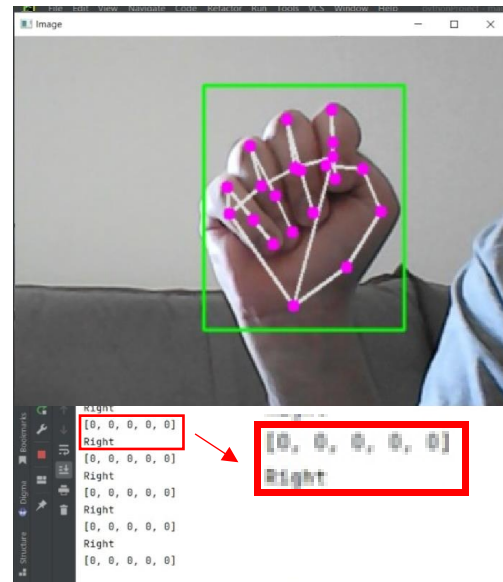


Fig. 11 - *PyCharm* reconhecendo a mão fechada e informando vetor com valores 0 para retorno.

Fonte: Autores, 2022.

No recorte em vermelho é destacado o vetor de valores 0 e a mão direita (*Right*) identificados. Na Figura 12, é mostrado o resultado esperado do movimento.



Fig. 12 - Resposta da mão ao receber vetor de valores 0.

Fonte: Autores, 2022.

## VI. TESTES E RESULTADOS

Após execução de testes pode-se concluir alguns casos específicos em que o reconhecimento de vídeo pode apresentar erros.

### 1. Interpretação ambígua

O primeiro caso observável é quando um dedo se encontra parcialmente dobrado, de forma que, o software não consegue atribuir um valor correto para a posição, pois possui apenas 0 e 1 como opções, aberto ou fechado. Por se tratar de uma visualização em tempo real, o código ao interpretar tanto um quanto o outro pode apresentar os dois movimentos entre alterações súbitas.

### 2. Distância da Captura

Este caso acontece quando a mão que está sendo analisada se encontra muito distante da câmera que realiza o reconhecimento, desta forma, a análise apresenta erros, pois não é possível identificar com precisão a posição e movimento dos dedos. A distância calculada entre os pontos chave diminui e por se tratar exatamente do parâmetro utilizado para detectar movimento, acaba por gerar interpretações equivocadas.

### 3. Múltipla Entrada

Apenas uma mão pode ser usada como parâmetro no reconhecimento. Quando mais de uma mão é capturada pela câmera, o software não consegue determinar qual deverá ser analisada. Neste caso, também serão apresentados erros, pois não é possível determinar qual mão é a “correta” para análise e ignorar outras.

## VII. CONCLUSÕES

Durante o desenvolvimento, foram encontradas diversas dificuldades. A impressão 3D do braço levou um tempo considerável até estar totalmente completa. Toda a estrutura do braço foi feita em impressão 3D através da universidade, porém, na primeira demanda de peças foi impresso com um material e na segunda demanda foi feita com outro material.

Além da diferença de resistência dos materiais, isso também acarretou grandemente em um atraso no desenvolvimento do projeto e na realização dos testes.

O que gerou mais trabalho foi a montagem física do projeto e durante a etapa de programação, ao executar o software com versões diferentes, ocasionou a maior parte dos erros, por isso foram escolhidas versões específicas no desenvolvimento. Já com o Arduino não ocorreram muitos imprevistos.

Entre estes fatores, também foi necessário readquirir servo motores danificados, comprar materiais necessários e outros para suporte visando manter o sistema conservado evitando avarias e organizado para o desenvolvimento e apresentação do projeto. O tempo de entrega de algumas peças foi um fator que também teve impacto na realização dos testes, além das dificuldades externas que giram em torno de conciliar tempo para conseguir efetivamente desenvolver o projeto.

Todavia, pode-se ter a conclusão de que o trabalho teve o principal objetivo alcançado: explorar alternativas para realizar reconhecimento de movimento por imagem e reconhecimento de voz para reprodução de movimentos de braço robótico. Tornando possível a execução de movimentos remotamente visando uma segurança maior para o ser humano ao realizar atividades necessárias que oferecem um risco à saúde dele e do próximo.

## REFERÊNCIAS

- [1] InMoov. Disponível em: <<https://inmoov.fr/>> Acesso em mar. 2022.
- [2] Arduino Uno. Disponível em: <<https://www.arduino.cc/en/software>>. Acesso em mar. 2022.
- [3] Voice Recognition V3. Disponível em: <<https://github.com/elechouse/VoiceRecognitionV3>>. Acesso em mar. 2022.
- [4] Pycharm. Disponível em: <<https://www.jetbrains.com/pycharm/>>. Acesso em abr. 2022.
- [5] Documentação CVZone. Disponível em: <<https://github.com/cvzone/cvzone>>. Acesso em mar. 2022.
- [6] Eletrogate. Disponível em: <[https://www.eletrogate.com/servo-towerpro-mg995-metalico-180?utm\\_source=Site&utm\\_medium=GoogleMerchant&utm\\_campaign=GoogleMerchant&gclid=CjwKCAiAmuKbBhA2EiwAxQnt7wKUNkwe8r0m\\_19tvgl9OHTctFu\\_t0I9LwUqi7dj96yyJy4NK1NehBoCkdWQAvD\\_BwE](https://www.eletrogate.com/servo-towerpro-mg995-metalico-180?utm_source=Site&utm_medium=GoogleMerchant&utm_campaign=GoogleMerchant&gclid=CjwKCAiAmuKbBhA2EiwAxQnt7wKUNkwe8r0m_19tvgl9OHTctFu_t0I9LwUqi7dj96yyJy4NK1NehBoCkdWQAvD_BwE)>. Acesso em ago. 2022.
- [7] Baú da Eletrônica. Disponível em: <<https://www.baudaeletronica.com.br/media/catalog/product/cache/1/image/800x/9df>>. Acesso em ago. 2022.
- [8] Elechouse. Disponível em: <<https://www.elechouse.com/product/speak-recognition-voice-recognition-module-v3/>>. Acesso em mar 2022.
- [9] Casa da Robótica. Disponível em: <<https://www.casadarobotica.com/placas-embarcadas/arduino/placas/placa-uno-r3-smd-atmega328-sem-cabo>>. Acesso em abr. 2022.
- [10] TinkerCad. Disponível em: <<https://www.tinkercad.com/>>. Acesso em jul. 2022
- [11] Documentação OpenCV. Disponível em: <<https://docs.opencv.org/2.4.13.7/>>. Acesso em mar. 2022.
- [12] Stack Overflow. Disponível em: <<https://pt.stackoverflow.com/>>. Acesso em ago. 2022.
- [13] Introdução a Robótica. Disponível em: <<https://www.amazon.com.br>>. Acesso em jul. 2022
- [14] Nilo Ney Coutinho Menezes, Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes. Disponível em: <<https://www.amazon.com.br/>>. Acesso em jul. 2022.
- [15] Faverin, Victor da Acidentes de trabalho com máquinas causaram mais de 25 mil amputações. Disponível em: <<https://revistacipa.com.br/acidentes-de-trabalho-com-maquinas-causaram-mais-de-25-mil-amputacoes/>>. Acesso em: 25 de set. 2022
- [16] Detalhamento de peças impressas do InMoov. Disponível em: <[https://inmoov.fr/inmoov-stl-parts-viewer/?bodyparts=Forearm-and-Servo-Bed&doing\\_wp\\_cron=1668978260.7184009552001953125000](https://inmoov.fr/inmoov-stl-parts-viewer/?bodyparts=Forearm-and-Servo-Bed&doing_wp_cron=1668978260.7184009552001953125000)>. Acesso em mar. 2022.

## APÊNDICE

- [1] Github: <https://github.com/Rafaelxndre/TCC-Braco-Robotico>