

# Sistema adaptativo para teleoperação de uma base móvel através de reconhecimento gestual

Dieisson Martinelli<sup>1</sup>, João Alberto Fabro<sup>1</sup>, André Schneider de Oliveira<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Avenida 7 de Setembro, 3165 – 80230-901  
Curitiba – PR – Brasil

dmartinelli@alunos.utfpr.edu.br, {fabro, andreoliveira}@utfpr.edu.br

**Abstract.** *This article presents the development of an adaptive system for teleoperating a mobile base coupled to a robotic arm with three degrees of freedom, through gesture capture from an RGB camera. This project uses the holistic pipeline of the MediaPipe framework to capture 2D body points of the operator through RGB images, and two algorithms are developed through this framework. The first algorithm is responsible for extracting features of the operator performing the movement required to perform a certain movement process. The second algorithm is responsible for using the data collected from the operator's body at the time of the process and identifying, through classification, the movement requested by the operator. After the movement classification, the key point position is calculated, which, through techniques proposed by the algorithm of this work, results in a value of 0 to 100 of activation of the required movement. This value goes through a fuzzy control system, which will emit the robot's movement. The tests are performed with 20 volunteer operators with the goal of following a trajectory and picking up/delivering an object.*

**Keywords:** *MediaPipe Framework, Teleoperation, Human Robot Interface, ROS*

**Resumo.** *Este artigo apresenta o desenvolvimento de um sistema adaptativo para teleoperação de uma base móvel acoplada a um braço robótico com três graus de liberdade, através de captura de gestos a partir de uma câmera RGB. Este projeto utiliza o pipeline holístico do framework MediaPipe para capturar pontos 2D do corpo do operador através de imagens RGB e dois algoritmos são desenvolvidos através deste framework. O primeiro algoritmo é responsável por extrair características do operador que realiza o movimento solicitado para executar um determinado processo de movimento. O segundo algoritmo é responsável por utilizar os dados coletados do corpo do operador no momento do processo e identificar, por meio de classificação, a movimentação solicitada pelo operador. Após a classificação do movimento, é calculada a posição do ponto chave, que, por meio de técnicas propostas pelo algoritmo deste trabalho, resulta em um valor de 0 a 100 de ativação do movimento requerido. Este valor passa por um sistema de controle fuzzy, que emitirá o movimento do robô. Os testes são realizados com 20 operadores voluntários com o objetivo de seguir uma trajetória e recolher/entregar um objeto.*

**Palavras-Chaves:** *Framework MediaPipe, Teleoperação, Interface Humano Robô, ROS.*

Este trabalho é o resumo da dissertação de mestrado intitulada de "Sistema adaptativo para teleoperação de base móvel através de reconhecimentos gestuais", orientado pelo professor André Schneider de Oliveira e Coorientado pelo professor João Alberto Fabro defendida no Programa de Pós-graduação em Engenharia Elétrica e Informática Industrial, da Universidade Tecnológica Federal do Paraná em 06 de maio de 2022. Esse documento tem por finalidade apresentar a dissertação de mestrado ao Concurso de Teses e Dissertações em Robótica (CTDR-2022), para concorrer na categoria de melhor dissertação de mestrado. O documento de dissertação completo pode ser acessado em <http://repositorio.utfpr.edu.br/jspui/handle/1/28785>. Esse trabalho foi contribuição direta para dois congressos [Martinelli et al. 2019] e [Martinelli et al. 2020]. O resultado deste trabalho encontra-se submetido em uma revista internacional, Qualis A1. Proporcionou o avanço e aprimoramento para técnicas de multirrobôs que resultou em uma submissão em um congresso internacional e seu resultado pode ser observado no link <https://youtu.be/10cVwcjPYOE>.

## 1. Introdução

O termo teleoperação se refere a uma técnica onde um operador humano controla um veículo ou um sistema à distância [Fong and Thorpe 2001]. Um sistema teleoperado é formado por duas partes, o módulo de controle e o telemanipulador. Tradicionalmente, a teleoperação é um método desenvolvido para aplicações onde a operação manual a bordo não pode ser usado por motivos de segurança.

Um sistema de teleoperação incorpora tanto atuadores para fornecer as informações do local remoto quanto sensores para codificar as ações de controle humano. Existem diversas abordagens de teleoperação, as mais tradicionais são de interface de envio de comandos que forçam o operador a usar equipamentos específicos para se comunicar um robô teleoperado, como *joysticks* táteis [Krishnasamy 2020], *cybergloves* [Almeida et al. 2019] e capacetes de captura de impulso cerebral [Galán et al. 2008].

A utilização de sistemas de visão computacional para extração de pontos-chaves do corpo humano já vem sendo implementadas há algum tempo e existem inúmeras pesquisas nesta área. Uma destas pesquisas é o uso desta técnica para teleoperação de diferentes tipos de robôs móveis.

Os trabalhos atuais em sua grande maioria utilizam estas técnicas associadas a movimentos estáticos, ou seja, os gestos são registrados previamente para um número fixo de operações. Essas operações contem uma rotina de trabalho para o robô, que a executará até o momento em que uma nova operação é requerida pelo usuário [Zhang et al. 2019, Qi et al. 2021, Christensen 2020].

Para solucionar os problemas das abordagens tradicionais de *interface* de movimentação e para propor melhorias no desempenho e novas abordagens que utilizam visão computacional, este trabalho propõe o uso de uma *interface* interativa e adaptável que utiliza pontos-chaves do corpo todo capturados de uma câmera colorida simples(RGB) para enviar comandos ao robô teleoperado. Dessa forma, nenhum equipamento físico adicional ou equipamento físico conectado ao operador se faz necessário, permitindo o operador fazer uso apenas de seu próprio corpo para enviar comandos para o robô.

## 1.1. Objetivos

O objetivo geral desse projeto é desenvolver uma abordagem adaptativa para envio de comandos de movimentação para um robô, recorrendo à técnicas de *deep learning* que reconhecem pontos-chaves no corpo humano. Estes pontos serão utilizados para efetuar um treinamento através de um classificador e posteriormente utilizados para identificar e executar o comando solicitado pelo operador, para permitir a teleoperação de sistemas robóticos de forma adaptativa e intuitiva.

## 2. Abordagem

Esta seção apresenta a proposta do projeto e é separada em 4 subseções demonstrando cada etapa. As seções apresentam a base móvel utilizada para os testes, a principal tecnologia utilizada para a abordagem desse projeto, técnicas desenvolvidas para a extração de características e a abordagem de movimentação desenvolvida.

### 2.1. Sistema Veículo-Manipulador

A base móvel utilizada para este projeto é a Pioneer 3-AT. O robô se conecta a um computador embarcado Intel NUC modelo 5I5RYH através de uma interface serial-USB permitindo sua conexão a um sistema ROS. A base também possui um *YDlidar X4*[YDlidar 2020], que é um sensor de distância de baixo custo, capaz de fazer medições de distâncias em 360°, na faixa de 12 centímetros até 10 metros com precisão de dez milímetros. Em sua camada superior foi acoplado o braço robótico 3R, que passou por um processo de *retrofit*.

O modelo original do manipulador utilizado é um Braço Robótico *Beckman Coulter ORCA*, um manipulador planar que possui três juntas rotacionais. Para que a estrutura respondesse aos comandos de posição de juntas desejados utilizando a estrutura ROS, foi realizado um processo de *retrofit*, onde os motores e *encoders* foram conectados a microcontroladores que permitem a programação aberta do mecanismo.

### 2.2. MediaPipe ROS Package

Para a utilização do *MediaPipe*[LLC 2020a] neste projeto, foi desenvolvido um pacote ROS do qual retornaria as informações coletadas no *framework* do mediapipe através de mensagens para tópicos ROS, para que sejam acessados por outros nós do processo dessa abordagem. Este pacote tem como finalidade converter o *MediaPipe Holistic* [LLC 2020b] em um pacote ROS para que possa ser utilizado por profissionais e entusiastas de robótica.

As mensagens são os dados lidos e escritos nos tópicos ROS. O pacote irá ler a imagem capturada pela câmera RGB, processá-la e publicar os resultados em dois tópicos, um contendo somente os *landmarks*, ou seja, os pontos que correspondem as partes do corpo detectados, e outro contendo a imagem com os *landmarks*.

### 2.3. Algoritmo de Extração

O algoritmo de extração de características tem como funcionalidade principal a criação de um *dataset* que contém informações de classe ativada, posição dos pontos do segmento de maior ativação, posição final e posição inicial do ponto-chave de maior ativação dentro do segmento.

O processo de extração de características passa por dois blocos de funções importantes, a escolha do segmento e a escolha do ponto de maior ativação dentro desse segmento. Para encontrar o segmento de maior ativação é utilizado o método que calcula a média de locomoção linear através da comparação entre a posição inicial do ponto-chave com sua posição anterior. Uma média entre a locomoção linear dentro de um segmento é realizada e dessa forma, cada um dos segmentos possui um valor de média de locomoção linear durante o processo de extração. O segmento que obtiver a maior média de locomoção durante o tempo de execução é escolhido como ativador da classe.

Para encontrar ponto de maior ativação dentro do segmento ativado foi definido um ponto de referência para cada um dos segmentos. Ambos os segmentos das mãos utilizaram o ponto do pulso como ponto de referência. O segmento da boca utilizou o ponto central do lábio superior. Quanto ao segmento do rosto, foi criado um ponto fictício central entre a posição do ombro esquerdo e ombro direito do operador.

Para a definição do maior ponto de ativação é feito o cálculo de diferença linear entre cada um dos pontos do segmento para o seu ponto de referência durante o tempo de extração de características.

As informações necessárias do ponto de maior ativação são sua posição inicial e final durante a extração. A posição inicial e final são definidos ou em X ou em Y, considerando em qual eixo houve uma maior movimento linear.

Durante o tempo de execução todos os pontos-chaves do segmento utilizado são armazenados no *dataset*. Estes dados serão utilizados posteriormente por um classificador, cujo objetivo será identificar a posição que indica o início da ação de uma determinada classe.

## **2.4. Técnica de Teleoperação**

Para poder executar o algoritmo de movimentação, o classificador SVM deve inicialmente passar por um processo de treinamento. Uma parcela de 33% do *dataset* é utilizado como base de teste, enquanto 66% são utilizados na etapa de treinamento.

Após isso o algoritmo de movimentação espera a captura de movimento do operador para classificar os dados enviados, em tempo de operação, utilizando o classificador treinado.

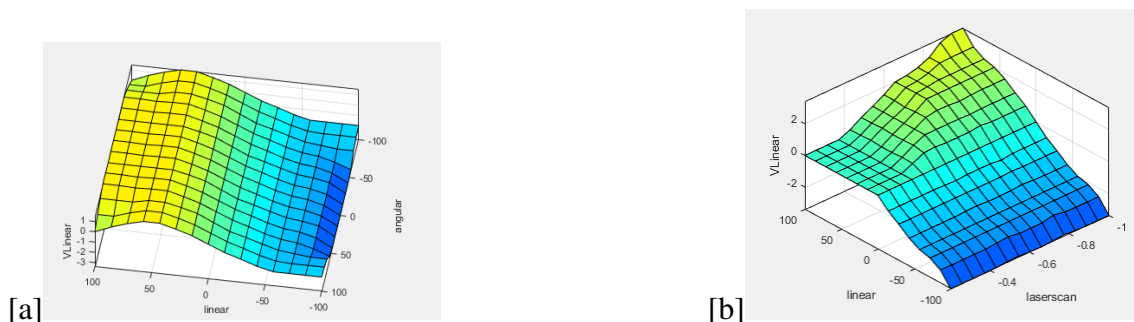
O sistema de classificação permite identificar se alguma classe está sendo ativada pelo operador. Caso a predição de alguma classe atinja um valor maior que 75%, o algoritmo desenvolvido entende que aquela classe está sendo requisitada pelo operador.

Com a informação da classe, é então verificado em qual eixo o ponto de maior ativação está atuando, ou seja, se está atuando no eixo Y ou eixo X.

O ponto de maior ativação tem uma variação de 0 a 100%. Essa variação é calculada dado a sua posição inicial e final.

Possuindo as informações de classe ativada e seu valores de ativação variando de 0 até 100, chega-se a última parte do algoritmo, o sistema de controle fuzzy.

As regras fuzzy são utilizadas para enviar comandos de velocidade linear e angular do robô de forma suave, não executando movimentos bruscos ao fazer uma curva, por exemplo. É utilizado também para evitar possíveis impactos com objetos em sua frente



**Figura 1. Gráficos apresentando as superfícies de controle, relacionando entradas (eixos "linear", "angular" e "distância") às saídas de Velocidade Linear (VLinear) aplicadas à base móvel.**

e para ativação do braço robótico. Ele conta com três entradas, sendo elas a Valor Linear, Valor Angular e a informação de Distância de objetos no ambiente, e três saídas, Velocidade Linear e Velocidade Angular que de fato será executada pelo robô móvel e a Ativação, responsável por iniciar as operações com o braço robótico.

a saída da Velocidade Linear sofre influência da entrada Angular e da Distância. Isso ocorre para que em momentos que a base recebe uma solicitação de movimentação para esquerda ou direita, a velocidade seja diminuída caso esteja muito rápida, para que o usuário consiga realizar a curva com maior precisão. No caso da Distância, um objeto próximo é identificado, através do laserscan, e sua velocidade é reduzida para melhorar o desempenho de movimentação do operador, evitando que colisões possam acontecer. A entrada Distância não influencia nas saídas de Velocidades Lineares negativos, pois o sensor YDlidar não identifica objetos na parte traseira do robô móvel. Porém, a influência da entrada Angular ainda ocorre para Velocidades Lineares negativos. Para um melhor entendimento foi gerada a Figura 1 contendo os gráficos de superfície das regras fuzzy onde é possível observar melhor a estratégia.

A saída Ativação é resultante da entrada de Distância, ou seja, o valor de ativação aumenta conforme o robô se aproxima de um objeto. No momento que o valor de ativação chegar no máximo a base móvel não se movimentará para frente deixando o controle total para o manipulador robótico, possibilitando a base móvel se movimentar apenas para direita, esquerda e para trás.

Para a teleoperação do braço robótico utilizou-se a biblioteca em Python chamada *TinyIK* [Lanius 2021]. Esta biblioteca permite a utilização de cinemática direta e inversa para controlar o braço robótico.

Primeiramente foi definida a posição inicial da ferramenta no espaço cartesiano com valores fixos de 0,25 mm para "x" e 0,16 mm para "y". A junta que controla a posição do punho ficará orientada a zero graus no eixo Z em relação ao sistema de coordenadas do mundo, portanto o manipulador ficará sempre paralelo ao chão. Para isso, a soma dos ângulos das juntas é equivalente à orientação do punho, portanto para este ficar orientado a zero graus, o ângulo da junta 3 é dado conforme Equação 1.

$$\theta_3 = - \left( \sum \theta_2 \theta_1 \right) \quad (1)$$

$\theta_1$  representa o valor angular da junta do ombro,  $\theta_2$  representa o valor angular da junta do cotovelo. A soma destes dois ângulos é multiplicada por -1 resultando no  $\theta_3$ , o ângulo da junta do punho.

Os valores obtidos através dos gestos dos operadores, os quais variam 0 a 100, foram convertidos para uma faixa de 0 a 0.545 mm, que é valor máximo de definido para o alcance das juntas em x e y no plano cartesiano, para as configurações de teleoperação do braço robótico.

A distância definida de locomoção entre a posição ponto atual no espaço cartesiano e a nova posição para realização da trajetória foi de 0.01 mm. Desta forma, o robô se desloca 0.01 mm por iteração até alcançar a nova posição requerida pelo operador. Para obter a distância entre um ponto e outro da posição solicitada pelo operador foi utilizado (a relação trigonométrica conforme Equação 2.

$$pm = \sqrt{(\Delta x * \Delta x) + (\Delta y * \Delta y)} \quad (2)$$

Onde  $\Delta x$  e  $\Delta y$  são representados pela diferença entre a posição do ponto atual e a posição do ponto requerido em x e y.

A cada iteração é verificado se a distância entre a posição do ponto solicitado pelo operador é maior que a distância mínima de locomoção definida. Caso essa distância seja maior, é feito o cálculo trigonométrico para obter a posição no espaço cartesiano que deve ser adotada pelo braço robótico através da Equação 3 e da Equação 4.

$$x = (\sin(x_a - x_i)) p (\cos(\text{atan2}(\Delta y, \Delta x))) + x_a \quad (3)$$

$$y = (\sin(y_a - y_i)) p (\sin(\text{atan2}(\Delta y, \Delta x))) + y_a \quad (4)$$

$x_a$  e  $y_a$  representam respectivamente as posições x e y atuais da ponta braço robótico. Enquanto  $x_i$  e  $y_i$  representam as posições solicitadas pelo operador. A variável  $p$  é a distância definida de locomoção nas configurações.

Para não permitir que o braço robótico chegue muito perto da estrutura de aço da base móvel foi definido o valor mínimo de posição de x em 0,25 mm.

Após isso, é utilizado funções da biblioteca *TinyIK* para realizar a cinemática inversa e obter os ângulos das juntas do ombro e cotovelo para realização da operação.

O movimento de abrir e fechar a garra foram convertido de 0 a 100, resultante da movimentação do operador, para completamente aberto e completamente fechado, não sofrendo nenhuma limitação para o processo.

### 3. Validação

Os experimentos deste trabalho foram realizados por 20 participantes voluntários de idade, gênero, nível de estudo e outras características distintas. Estes participantes foram submetidos ao processo de extração de características para definição dos movimentos e a um processo de realização de trajetória dentro de um cenário.

Nesta seção são desenvolvidas as análises do extrator de características e sua classificação, mostrando dados de desempenho de treinamento e desempenho em

operação. As análises da operação utiliza 60 registros de execuções feitas por 20 operadores voluntários e tem como objetivo demonstrar se características específicas destes usuários afetam o resultado final do sistema proposto.

### 3.1. Análise do Processo de Extração de Dados

O processo de extração de características foi realizado com algumas restrições exigidas pelo método proposto para uma maior precisão. O operador deve se posicionar a uma distância de máximo 70 cm da câmera RGB que está voltada para ele. Como toda a coleta de pontos se dá em um universo bidimensional o MediaPipe não nos dá a informação de distância do ponto, sendo necessário manter uma distância média da câmera RGB para melhor precisão de extração de características para o classificador e para o cálculo de posição inicial e final do ponto de ativação, que será posteriormente utilizado para enviar comandos de 0 a 100 para o Sistema Veículo-Manipulador.

Com estas informações o processo de extração de características se deu início e o resultado pode ser observado na Tabela 1. Para cada classe foi necessário a realização da extração de característica 3 vezes, levando em média 2.5 minutos por operador.

**Tabela 1. Porcentagem de cada segmento usado para cada uma das classes de movimento.**

-	Boca	Cabeça	Mão Direita	Mão Esquerda
<b>Forward</b>	15%	15%	25%	45%
<b>Backward</b>	10%	10%	20%	60%
<b>Left</b>	0%	0%	50%	50%
<b>Right</b>	0%	0%	70%	30%
<b>Up/Down</b>	0%	60%	35%	5%
<b>Effector</b>	55%	10%	25%	10%

É interessante notar que os segmentos predominantes para controlar a direção esquerda e direita foram os segmentos das mãos. Mas no restante das classes os segmentos foram diversificados permitindo assim uma melhor análise do sistema de movimentação em questão de adaptação do operador, visto que, apesar da predominância do uso do segmento das mãos nas classes de direita e esquerda o restante da estrutura de teleoperação é diversificada para cada um dos operadores, contendo diferentes pontos de ativação.

### 3.2. Análise de Classificação

Para o treinamento do classificador 33% da base de dados foi utilizada como base de testes e o restante utilizado como treinamento para cada um dos operadores. Esta configuração sempre retornou um valor de 100% de acurácia durante o treinamento.

Após o treinamento foi realizada a validação do classificador durante a operação, e para isso um experimento foi conduzido. A ideia foi contabilizar o reconhecimento do classificador no momento que o operador deseja executar um movimento. Para isso, é primeiro necessário saber qual dos movimentos o operador deseja executar e verificar se a classe correspondente a esse movimento estava sendo realmente ativada. Para isso foi questionado ao operador qual movimento ele iria ativar e contabilizado a porcentagem do reconhecimento da classe durante o movimento. Foi utilizado neste experimento a função *predict\_proba* do pacote *Sklearn*<sup>1</sup>, essa função fornece as probabilidades para a classe

<sup>1</sup>Biblioteca de aprendizado de máquina de código aberto para a linguagem de programação Python.

em forma de matriz. O número de probabilidades para cada linha é igual ao número de classes. Dessa forma conseguimos medir a probabilidade de uma classe ser o resultado de saída para os pontos classificados.

O experimento foi realizado com os 20 participantes com as diferentes configurações feitas por cada um deles. Os resultados podem ser observados na Tabela 2.

**Tabela 2. Classificação média das classes durante a operação dos operadores voluntários.**

Classificador de média geral	
Classe	Média
Foward	85,96%
Backward	84,71%
Right	90,76%
Left	86,36%
Up/Down	92,34%
Effector	86,29%
<b>Média geral</b>	<b>87,74%</b>

Em média o classificador conseguiu obter um resultado de 87,74% em operação, a classe "Backward" obteve o menor valor de acurácia, 84,71% e o maior valor obtido foi de 92,34% pela classe "Up/Down". Isso demonstra que, mesmo durante a execução, não houveram casos de falsos negativos e falsos positivos e o classificador concluiu o seu objetivo. Em 20 configurações diferentes obteve resultados maiores que 80% para todas as classes. Durante o acompanhamento destes testes realizados neste trabalho, foi questionado aos operadores voluntários se em algum momento o sistema não reconheceu o verdadeiro movimento que estava sendo solicitado. Como resposta, nenhum operador voluntário identificou um momento de falha de reconhecimento.

### 3.3. Análise da Operação

Para esta análise foi requisitado a execução de determinadas tarefas para que os operadores voluntários passassem por uma rotina. Esta rotina deveria ser realizada três vezes, de modo a identificar a adaptabilidade do usuário ao sistema de teleoperação. Algumas características de cada operador foram também analisadas, visando identificar possíveis fatores que poderiam dificultar a utilização do sistema.

A operação consiste no operador teleoperar o Sistema Veículo-Manipulador através dos comandos registrados por um cenário, onde ele tem como objetivo primário seguir uma linha até o objeto de coleta e então levar o objeto até o ponto de destino. A Figura 2, ilustra este processo.

A acurácia final do processo de operação é resultando da média de 3 processos: seguir a linha, coletar o objeto e entregar o objeto no ponto de destino. O processo de seguir a linha foi calculado com base no tempo de execução da tarefa. Durante todo tempo de execução foi calculado o quanto desse tempo o operador se encontrava dentro da trajetória definida. O processo de coleta e entrega de objeto era definido como um valor booleano, entregou ou não entregou. O resultado desta operação deu origem a Tabela 3.

Ao observar estes dados é possível notar que apesar de a acurácia obter um valor relativamente baixo de 67.66%, a porcentagem média de melhoria entre os testes foi de



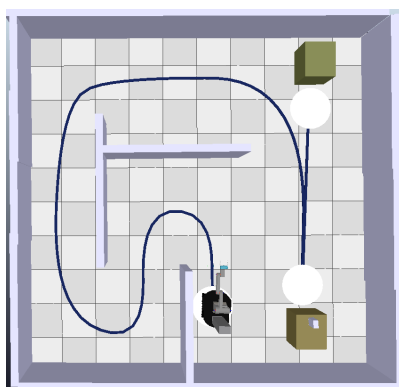


Figura 2. Ilustração do processo de operação realizado pelos operadores.

Tabela 3. Informações de todos os operadores voluntários.

Voluntário	Idade	Gênero	Nível de Estudo	Área de Trabalho	Habilitação	Tempo(s)	Melhoria (%)	Acurácia (%)
1	18	Feminino	Ensino Médio	Operacional	Sim	672,7	48,77%	73,65%
2	18	Feminino	Graduação	Operacional	Sim	386,3	51,44%	74,81%
3	22	Feminino	Graduação	Biológicas	Sim	462,0	30,20%	73,40%
4	22	Masculino	Ensino Médio	Operacional	Sim	353,3	5,00%	91,60%
5	22	Masculino	Graduação	Exatas	Não	531,0	7,00%	91,87%
6	23	Feminino	Graduação	Humanas.	Sim	469,7	56,34%	66,67%
7	24	Masculino	Ensino Médio	Operacional	Sim	399,0	59,39%	63,45%
8	24	Masculino	Graduação	Exatas	Sim	353,3	5,00%	91,60%
9	25	Masculino	Graduação	Operacional	Sim	486,3	1,83%	78,04%
10	25	Masculino	Graduação	Humanas.	Sim	442,0	3,23%	81,31%
11	27	Feminino	Graduação	Humanas.	Sim	419,3	56,60%	63,83%
12	30	Masculino	Graduação	Exatas	Sim	406,3	10,98%	72,08%
13	40	Feminino	Ensino Médio	Operacional	Sim	1032,7	49,95%	51,15%
14	41	Masculino	Ensino Médio	Operacional	Sim	586,7	56,65%	66,42%
15	48	Feminino	Ensino Médio	Operacional	Sim	383,3	41,50%	47,83%
16	51	Feminino	Básico	Operacional	Não	958,7	10,10%	73,20%
17	54	Feminino	Básico	Operacional	Não	794,3	10,59%	31,25%
18	58	Feminino	Tecnico	Biológicas	Não	1058,0	16,70%	74,57%
19	59	Masculino	Ensino Médio	Operacional	Sim	493,3	57,50%	56,50%
20	65	Feminino	Básico	Operacional	Não	1307,3	6,59%	30,08%
					<b>Média Final</b>	<b>599,78</b>	<b>29,27%</b>	<b>67,66%</b>

29,27%. Recordando que apenas foram realizados 3 testes por voluntário e nestes 3 testes todos os voluntários obtiveram uma porcentagem de melhora, demonstrando dessa forma que a adaptação ao sistema de teleoperação ocorreu.

Para uma melhor análise, são utilizadas as cinco primeiras colunas desta tabela para estimar o quanto essas informações impactaram no resultado final obtido. Com isso é possível fazer posteriormente uma análise geral da estratégia proposta.

Para realizar a avaliação do quanto a Idade, Gênero, Nível de Estudo, Área de Trabalho e a informação se o operador dirige ou não interferem no resultado final foi primeiramente feito uma média dos resultados obtidos por estas colunas e realizado algumas análises.

Quando analisado os dados da idade foi possível perceber que os resultados foram bem diversificados, mostrando que, a informação de idade não interfere na habilidade do voluntário em operar o sistema.

Quanto a informação de gênero demonstra que, a média de acurácia tem uma diferença de 16,58% entre o gênero feminino e masculino. Porém, a porcentagem de melhora entre o gênero feminino demonstrou ser superior ao masculino, tendo uma diferença de 11,48%. Demonstrando que, apesar do gênero feminino apresentar uma

acurácia menor em comparação ao gênero masculino, sua porcentagem de melhora é maior ao longo de execuções, mostrando que esses usuários se adaptaram ao sistema ao longo das execuções.

Foi analisado que quanto maior o nível de estudo do usuário, maior foi sua média de acurácia final. Porém, também é possível observar que operadores com Ensino Médio obtiveram a maior porcentagem média de melhora durante o processo. Demonstrando novamente que usuários que não obtiveram uma boa acurácia nas primeiras execuções se adaptaram e melhoraram ao longo das execuções. Assim como observado na análise da área de trabalho de cada operador, sendo possível observar que operadores da área de exatas obtiveram uma acurácia superior aos demais. Em contra ponto, sua porcentagem de melhora foi de apenas 7,66%. Por outro lado, operadores da área operacional obtiveram a menor acurácia média, mas a porcentagem de melhora entre os testes foi em média de 33,27%.

Em uma análise entre pessoas que possuem experiência em direção de veículos e as que não possuem, é possível observar que a acurácia final média foi muito semelhante entre ambos os casos, diferenciando apenas em 10%. Porém, é importante ressaltar que, pessoas sem experiência em direção obtiveram uma porcentagem de melhora entre os 3 testes 25,44% menor do que operadores com experiência, o que pode indicar que a falta de experiência com a direção pode influenciar no resultado.

Com estas informações foi calculado o coeficiente de variação entre o resultado final e cada um destes parâmetros anteriormente analisados. O coeficiente de variação da porcentagem de melhora também foi calculado. Quanto menor for o valor do coeficiente de variação, mais homogêneos serão os dados, ou seja, menor influência ele terá sobre o resultado final. Desta forma, é possível ter uma ideia se características específicas de cada pessoa interferem ou ajudam no resultado e quantificar este valor. Esse resultado será utilizado como uma forma de medir o quanto o sistema desenvolvido por este projeto se adapta a diferentes pessoas, com diferentes características. O resultado destas medições pode ser observado na Tabela 4.

**Tabela 4. Coeficiente de variação (CV) da acurácia média e melhora média obtida pelos operadores de cada característica analisada.**

-	CV Accuracy	CV Improvement
Idade	2,84%	4,74%
Gênero	0,72%	0,33%
Nível de Estudo	1,62%	1,91%
Área de Trabalho	0,77%	1,42%
Habilitação(S/N)	0,27%	1,75%

Com esta tabela podemos observar que características específicas do operador não influenciaram o processo de teleoperação devido aos baixos valores calculados. Estas características também não influenciam no processo de melhora durante as execuções. Dessa forma, é possível afirmar que o algoritmo se adapta de forma semelhante, a cada operador, independente das características específicas analisadas por este trabalho.

Os diferentes experimentos que avaliam a abordagem do sistema teleoperação visam determinar e quantificar o nível de adaptabilidade do sistema para diferentes operadores. Foram realizados um total de 60 testes reproduzidos por 20 operadores.

Durante todos os testes a taxa de comunicação entre o operador e o robô foi armazenada e classificada em uma média de 266 ms. Essa taxa representa o tempo da estratégia proposta identificar o movimento requerido pelo operador até o momento em que o robô de fato realiza o movimento requerido. A taxa é resultante da média de todos os operadores durante as 60 operações.

Devido a rápida taxa de resposta da estratégia proposta os operadores voluntários conseguiram realizar uma trajetória suave durante todo o percurso exigido pelos testes.

#### **4. Conclusão**

O objetivo deste trabalho foi desenvolver uma abordagem adaptativa para envio de comandos teleoperados para movimentação de um robô. Para isso, a proposta utilizou técnicas de *deep learning* que reconhecem pontos-chaves no corpo humano utilizados para o treinamento de um classificador utilizado para reconhecimento de classe de gestos na operação.

Para o desenvolvimento do Sistema Veiculo-Manipulador foi realizado um retrofit do manipulador robótico ORCA e o seu acoplamento a base móvel. Toda a estrutura foi desenvolvida para funcionamento através do sistema ROS. Experimentos foram realizados para verificar seu funcionamento sendo demonstrado que os erros ocorridos durante os experimentos foram de cunho físico do equipamento, devido a desgastes do tempo. Uma versão simulada do Sistema Veiculo-Manipulador foi desenvolvido para facilitar experimentos durante o desenvolvimento.

A validação deste trabalho se deu através de 60 execuções realizadas por 20 operadores voluntários com características e níveis de experiências diferentes. A execução contava com dois objetivos principais, seguir uma linha e coletar/entregar o objeto. Com este experimento foi possível extrair informações de acurácia de classificador para diferentes categorias de gestos e movimentos, informações de taxa de tempo de comunicação, influência de características específicas do operador no sistema, acurácia média entre os 3 testes realizados por cada operador e a informação de melhoria de cada operador durante os testes realizados.

A interface desenvolvida se mostrou adaptável a diferentes operadores com diferentes características, experiências e níveis de habilidade para teleoperar um robô móvel com um braço robótico de três graus de liberdade acoplado a estrutura, em uma tarefa de coleta e entrega de objeto. É possível observar através dos resultados que a adaptação do usuário ocorre de forma rápida, tendo uma melhoria média de 29,27% em apenas 3 execuções.

O sistema proposto demonstrou um ótimo resultado de classificação dos gestos mesmo com pouco conteúdo no *dataset*. A realização dos registros de gestos se mostrou de fácil entendimento para os operadores, que conseguiram realizar a classificação de todos os movimentos. Durante a execução o sistema obteve uma ótima acurácia de reconhecimento e a técnica desenvolvida para o envio de comandos se mostrou de fácil uso pelo operador.

Apesar de a média de acurácia da realização das tarefas ser de 67,66%, a média de melhoria do operador durante as execuções foi de 29,27%. Demonstrando que, apesar da dificuldade durante a primeira execução os operadores se adaptaram rapidamente ao

sistema, obtendo resultados melhores conforme seguiram as execuções, resultado esse que pode também ser observado através da comparação das trajetórias realizadas entre a primeira e terceira execução.

## Referências

- Almeida, L., Lopes, E., Yalçinkaya, B., Martins, R., Lopes, A., Menezes, P., and Pires, G. (2019). Towards natural interaction in immersive reality with a cyber-glove. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 2653–2658. IEEE.
- Christensen, S. L. H. I. (2020). Wavetofly: Control a uav using body gestures.
- Fong, T. and Thorpe, C. (2001). Vehicle teleoperation interfaces. *Autonomous robots*, 11(1):9–18.
- Galán, F., Nuttin, M., Lew, E., Ferrez, P. W., Vanacker, G., Philips, J., and Millán, J. d. R. (2008). A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots. *Clinical neurophysiology*, 119(9):2159–2169.
- Krishnasamy, B. (2020). Design of a new industrial proportional joystick manufactured with 3d printing.
- Lanius (2021). tinyik. <https://github.com/lanius/tinyik>.
- LLC, G. (2020a). Mediapipe. LLC, GOOGLE.
- LLC, G. (2020b). Mediapipe holistic. LLC, GOOGLE.
- Martinelli, D., Cerbaro, J., Fabro, J. A., de Oliveira, A. S., and Teixeira, M. A. S. (2020). Human-robot interface for remote control via iot communication using deep learning techniques for motion recognition. In *2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE)*, pages 1–6. IEEE.
- Martinelli, D., Sousa, A. L., Augusto, M. E., Kalempa, V. C., de Oliveira, A. S., Rohrich, R. F., and Teixeira, M. A. (2019). Remote control for mobile robots using gestures captured by the rgb camera and recognized by deep learning techniques. In *2019 Latin American Robotics Symposium (LARS), 2019 Brazilian Symposium on Robotics (SBR) and 2019 Workshop on Robotics in Education (WRE)*, pages 98–103. IEEE.
- Qi, W., Ovrur, S. E., Li, Z., Marzullo, A., and Song, R. (2021). Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network. *IEEE Robotics and Automation Letters*, 6(3):6039–6045.
- YDLidar (2020). ydlidar\_ros. [https://github.com/YDLIDAR/ydlidar\\_ros](https://github.com/YDLIDAR/ydlidar_ros).
- Zhang, W., Cheng, H., Zhao, L., Hao, L., Tao, M., and Xiang, C. (2019). A gesture-based teleoperation system for compliant robot motion. *Applied Sciences*, 9(24):5290.