



Universidade Federal de Pernambuco  
Centro de Informática

**Aprendizado de Máquina para  
Reconhecimento de Gestos Manuais  
Aplicado à LIBRAS (Linguagem  
Brasileira de Sinais)**

Disciplina: Introdução à Computação - IF668[1]

Estudantes	Eric Vinícius de Lima[3] Michel Leonidas Aleixo da Silva[2]
Professor	Adriano Lorena Inacio de Oliveira[9]
Horário	Terça-Feira - 10:00-12:00 AM

Recife, 24 de Outubro de 2021

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Ferramentas Usadas</b>	<b>2</b>
<b>3</b>	<b>Procedimento Experimental</b>	<b>2</b>
3.1	Coleta de Dados: . . . . .	3
3.2	Popular Banco de Dados de Treinamento: . . . . .	4
3.3	Treinando a Inteligência Artificial: . . . . .	4
<b>4</b>	<b>Análise de Resultados:</b>	<b>5</b>
<b>5</b>	<b>Conclusão</b>	<b>6</b>
<b>6</b>	<b>References</b>	<b>7</b>

# 1 Introdução

Após uma aula introdutória sobre Aprendizagem de Máquina ministrada pelo Professor Adriano L. I. Oliveira, foi solicitado um projeto com o objetivo geral de aplicar na prática os conhecimentos de Inteligência Artificial. Construindo um programa que envolvesse aprendizado de máquina com modelos baseados em Texto, Imagens, Números ou Sons, aplicáveis a qualquer situação real ou não.

A Linguagem Brasileira de Sinais - LIBRAS - é considerada oficial no país desde 2005 [13], porém, mesmo com mais de 9 milhões de deficientes auditivos (2010) [6], ainda não é ensinada nas escolas públicas, tornando a aprendizagem desta linguagem ainda mais difícil e, conseqüentemente, a comunicação com o deficiente auditivo.

Com isso, decidimos desenvolver uma aplicação que utiliza Aprendizado de Máquina capaz de julgar com precisão se o gesto feito com a mão é ou não uma vogal pela Linguagem Brasileira de Sinais, através de uma webcam.



Figura 1: Inclusão e acessibilidade de deficientes auditivos[12]

## 2 Ferramentas Usadas

Para este experimento, a linguagem de programação Python[11] foi utilizada em conjunto com os frameworks MediaPipe[5] e OpenCV[10], além da plataforma Machine Learning for Kids[8] para processamento de resultados com o Watson Assistant da IBM[7].



Figura 2: Ferramentas usadas no projeto

## 3 Procedimento Experimental

Inicialmente, foi estudado como funciona o reconhecimento de uma mão pelo computador. Normalmente podemos representá-lo computacionalmente como um grafo de 21 vértices. Para reduzir a complexidade, decidimos trabalhar com apenas 11 deles.

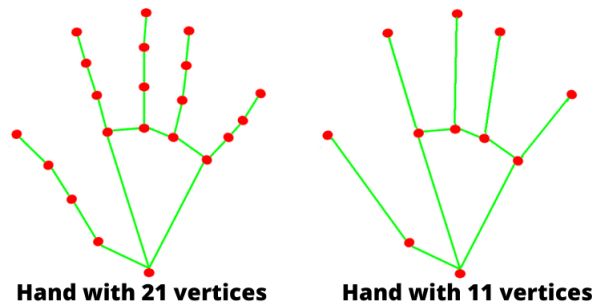


Figura 3: Mão com diferentes quantidades de vértices

Além disso, optou-se por julgar apenas as vogais e não todo o alfabeto em LIBRAS. O Machine Learning for Kids permite que apenas 10 rótulos funcionem com números. Cada vértice possui uma coordenada contendo 2 valores: eixo x e eixo y. Foi então introduzido um impedimento, pelo que decidiu-se reduzir ainda mais o número de vértices, agora para apenas 5. Os vértices mais relevantes foram então escolhidos, seguindo os seguintes gestos:



Figura 4: Vogais em LIBRAS com os principais vértices

Seguindo a nomenclatura de vértices adotada pelo Mediapipe, os vértices escolhidos foram: 0-wrist, 4-thumb-tip, 5-index-finger-mcp, 12-middle-finger-tip and 20-pinky-tip. Com isso, foi iniciada a elaboração do código do projeto, que foi dividido em 3 partes:

### 3.1 Coleta de Dados:

A primeira parte do projeto foi utilizar as bibliotecas mencionadas para coletar e tratar os dados. O OpenCV foi utilizado para reconhecer uma câmera e realizar algumas configurações de imagem. Em seguida, utilizamos algumas soluções presentes no Mediapipe, para detectar e reconhecer mãos, além de imprimi-las na tela. O MediaPipe trata a mão com 21 vértices, então tratamos esses vértices para pegar apenas 5 deles e remover o eixo z, pois não é necessário para o projeto.

Por fim, foi possível coletar, por meio de teclas pressionadas no teclado, os eixos x e y de cada um dos vértices escolhidos.



**The Input Image**

```
{
    'WRIST': (398.56807708740234,
              449.34757232666016),
    'THUMB_TIP': (211.23279571533203,
                  316.96200370788574),
    'INDEX_FINGER_MCP': (348.084716796875,
                         289.4392490386963),
    'MIDDLE_FINGER_TIP': (386.728515625,
                           136.52099132537842),
    'PINKY_TIP': (520.3791809082031,
                  215.13034343719482)
}
```

**The Output Data**

Figura 5: Coleta de Dados

### 3.2 Popular Banco de Dados de Treinamento:

Para nos comunicarmos com o Machine Learning for Kids, utilizamos sua API, na qual é possível solicitar a classificação dos dados e também o preenchimento do banco de dados de treinamento, e integrá-lo ao projeto através da biblioteca Python Request[4]. Para isso, foi criado outro módulo responsável por fazer essa comunicação, além de gerar uma chave de acesso API diretamente no Machine Learning for Kids, bem como realizar outras configurações, como a criação de rótulos para as vogais que serão preenchidas.

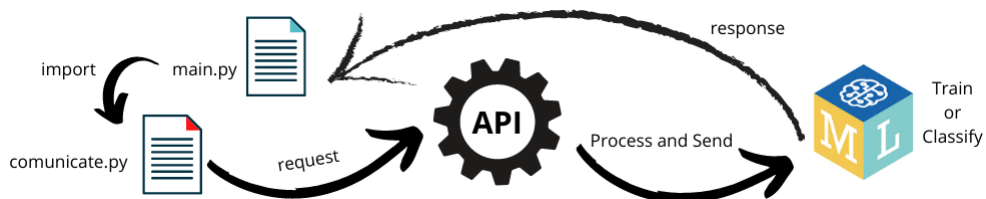


Figura 6: Diagrama da API

### 3.3 Treinando a Inteligência Artificial:

Todo o treinamento do computador foi realizado pelo Watson, que utiliza modelos de Rede Neural Convolutacional e de regressão linear para classificação. Assim, inicialmente foram adicionados 50 gestos para cada vogal, feitos por pessoas diferentes, com a mão direita e esquerda e em webcams diferentes, para gerar maior precisão nos resultados.



Figura 7: Diagrama do Treinamento

O processo de entrada de dados para o banco de dados de treinamento foi automatizado, agora sendo feito pressionando uma tecla do teclado, específica para cada gesto vocálico. E para facilitar a visualização dos vértices principais, optou-se por destacar esses pontos na janela da webcam além de imprimir nesta janela um texto com a o retorno da solicitação efetuada.

## 4 Análise de Resultados:

Durante a análise dos resultados, que ocorreu em 21/08/2021, o Machine Learning For Kids teve problemas para retornar o percentual de confiança, sempre apresentando 100% de confiança em todos os testes realizados (corretos e errados). Portanto, para verificar a real confiança desse juiz, foi utilizada a média geral, por meio do número de acertos e erros coletados para cada vogal. Então, os seguintes dados foram coletados:

Análise de Resultados - 08/21/2021				
Gesto	Testes Feitos	Acertos	Erros	Média de Acertos
A	20	12	8	60%
E	20	15	5	75%
I	20	18	2	80%
O	20	12	8	60%
U	20	17	3	85%

Tabela 1: Análise de Resultados

Percebeu-se que a maior taxa de erro foi com as vogais A e O, uma possível justificativa é que, graficamente, a localização destes pontos é semelhante. Além disso, os erros nas demais vogais ocorreram em maioria quando a mão estava a uma distância maior que um metro da webcam. Portanto, mais 50 gestos foram adicionados ao banco de dados de treinamento (totalizando 100 casos de exemplos por vogal), e novos resultados foram obtidos:

Nova Análise de Resultados - 08/22/2021				
Gesto	Testes Feitos	Acertos	Erros	Média de Acertos
A	30	28	2	93.3%
E	30	29	1	96.6%
I	30	30	0	100%
O	30	29	1	96.6%
U	30	30	0	100%

Tabela 2: Nova Análise de Resultados

Com esses resultados mais precisos e com um percentual de acertos acima de 90%, conclui-se que esta aplicação possui alta confiabilidade no julgamento de vogais da Língua Brasileira de Sinais feitas à mão.

## 5 Conclusão

O aprendizado da Linguagem Brasileira de Sinais continuará sendo um desafio não só para os deficientes auditivos, mas também para toda a população. No entanto, aplicativos como este, que são de domínio público, podem servir não apenas para aprender mais sobre o assunto, mas também para inspirar outras aplicações.

A aplicação já se mostrou capaz de cumprir sua função inicial - reconhecer vogais feitas com gestos pela LIBRAS -. Ele usa o Aprendizado de Máquina e, por isso, aprende a cada interação, o que o torna cada vez mais inteligente. Podendo ser utilizado na área de educação, tanto no uso do próprio da aplicação, quanto no ensino de LIBRAS ou no ensino de disciplinas relacionadas ao Aprendizado de Máquina, podendo também ser aprimorado adicionando outro modelo de aprendizado de máquina, adicionando mais vértices, ou até mesmo o fazer em outra linguagem de programação, tornando-o mais acessível para outros públicos.



## Referências

- [1] CIn. Introdução à computação - if668. Online; acessado em 20/08/2021.
- [2] Michel Leonidas Aleixo da Silva. Michel leonidas. Online; acessado em 20/08/2021.
- [3] Eric Vinícius de Lima. Eric de lima. Online; acessado em 20/08/2021.
- [4] Python Software Foundation. Requests: Http for humans. Online; acessado em 20/08/2021.
- [5] Google. Mediapipe. Online; acessado em 20/08/2021.
- [6] IBGE. Total de pessoas com deficiência auditiva. Online; acessado em 20/08/2021.
- [7] IBM. Ibm - watson. Online; acessado em 20/08/2021.
- [8] IBM. Machine learning for kids. Online; acessado em 20/08/2021.
- [9] System of Curriculum Lattes. Curriculum lattés adriano l. i.;de oliveira. Online; acessado em 20/08/2021.
- [10] OpenCV Org. Opencv. Online; acessado em 20/08/2021.
- [11] Python Org. Python. Online; acessado em 20/08/2021.
- [12] Isabela Palma. Handtalk. Online; acessado em 20/08/2021.
- [13] Planalto. Decreto nº 5.626. Online; acessado em 20/08/2021.