

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE INDAIATUBA

DR. ARCHIMEDES LAMOGLIA

CURSO DE TECNOLOGIA EM

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

CARLOS AUGUSTO HUBERT

MATEUS ROCHA PEREIRA

**ESTRUTURAÇÃO DE UMA BASE DE DADOS VOLTADA AO
TREINAMENTOS DE MODELOS DE RECONHECIMENTO
DE SINAIS DE LIBRAS**

INDAIATUBA

2021

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE INDAIATUBA

DR. ARCHIMEDES LAMOGLIA

CURSO DE TECNOLOGIA EM

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

CARLOS AUGUSTO HUBERT

MATEUS ROCHA PEREIRA

**ESTRUTURAÇÃO DE UMA BASE DE DADOS VOLTADA AO
TREINAMENTOS DE MODELOS DE RECONHECIMENTO
DE SINAIS DE LIBRAS**

Projeto de Trabalho de Graduação apresentado por Carlos Augusto Hubert e Mateus Rocha Pereira como pré-requisito parcial para a conclusão do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Faculdade de Tecnologia de Indaiatuba, elaborado sob a orientação do Prof. Sergio Donisete Clauss.

INDAIATUBA

2021

RESUMO

Estudos históricos destacam as grandes invenções da humanidade: a roda, a internet, o motor a vapor e a luz. Porém outras vertentes pontuam a fala como uma das mais importantes invenções da raça humana. Segundo pesquisas realizadas pelo IBGE (significado sigla), no Brasil existem mais de 10 milhões de pessoas com algum grau de deficiência auditiva, que em sua maioria fazem uso e fruto da Libras (Língua Brasileira de Sinais) como principal forma de expressão, sendo ela considerada a segunda língua oficial do país. Buscando uma visão tecnológica do uso da Libras como forma de comunicação, existe uma gama considerável de aplicações que auxiliam pessoas ouvintes a se comunicarem com deficientes auditivos, que proporcionam tradução instantânea da língua portuguesa para a língua de sinais, contudo o contrário não é verídico, hoje encontra-se um déficit de projetos voltados para a transcrição da Libras para a língua falada. Assim, este trabalho tem como pilar primordial a utilização de mapeamento de pontos de referência nas mãos visando não utilizar de bases de dados para a realização de treinamentos, identificando padrões e definindo conjuntos de posições dos pontos referidos, também com foco na utilização de recursos de captura de imagem básicos, como câmeras / webcams, garantindo uma solução de baixo custo de implementação. Com isso, almejando realizar a transposição da língua de sinais para o português.

Palavras-chave: Comunicação; Língua de Sinais; Estrutura de Dados; Visão Computacional.

FIGURAS

Figura 1: Configurações de mão extraídas do Dicionário Digital da INES	16
Figura 2: Configuração de Mão da Libras.....	17
Figura 3: Representação do parâmetro “movimento”	18
Figura 4: Representação do parâmetro “ponto de articulação”	18
Figura 5: Representação do parâmetro “orientação”	19
Figura 6: Representação do parâmetro “expressão”	20
Figura 7: Exemplo de Lista linear	23
Figura 8: Exemplo de Fila	24
Figura 9: Exemplo de Pilha	25
Figura 10: Exemplo de Árvore	26
Figura 11: Exemplo de Árvore de Especialização.....	26
Figura 12: Exemplo de Árvore de Composição	27
Figura 13: Exemplo de Árvore de Dependência	27
Figura 14: Exemplo de parte da estrutura do diretório da base de dados	28
Figura 15: Exemplo de instância da base de dados	29
Figura 16: Mapeamento da mão em tempo real usando o HandPose.....	30
Figura 17: Exemplo de estrutura para o reconhecimento de faces	31
Figura 18: Teste do sistema SignPic em iOS	32
Figura 19: Sistema para detecção e contagem de pessoas.....	33
Figura 20: Pontos de articulação da mão.....	37
Figura 21: Árvore da base de dados	38
Figura 22: OpenCV acessando câmera.....	39
Figura 23: MediaPipe mapeando pontos de referência	40

QUADROS

Quadro 1: Trabalhos relacionados	34
--	----

TABELAS

SUMÁRIO

INTRODUÇÃO	10
CAPÍTULO I	13
Fundamentação Teórica	13
1.1 Conceitos chave	13
1.1.1 História dos Surdos no Brasil	13
1.1.2 Língua Brasileira de Sinais (Libras) e sua fonologia	15
1.1.2.1 Configuração das mãos.....	16
1.1.2.2 Movimento	17
1.1.2.3 Pontos de articulação	18
1.1.2.4 Orientação da mão	19
1.1.2.5 Expressões	19
1.1.3 Processamento de Imagens e Visão computacional	20
1.1.3.1 Processamento de Imagens	20
1.1.3.2 Visão computacional	21
1.1.4 Estrutura de dados	21
1.1.4.1 Listas.....	22
1.1.4.2 Filas	23
1.1.4.3 Pilhas	24
1.1.4.4 Árvores	25
1.2 Trabalhos relacionados	27
CAPÍTULO II.....	35
Metodologia.....	35
2.1 Natureza da Pesquisa	35

2.2	Variáveis de análise	35
2.3	Definição e apresentação das Ferramentas para Análise	35
2.3.1	Python.....	36
2.3.2	OpenCV	36
2.3.3	MediaPipe.....	37
2.4	Experimento de Pesquisa.....	37
2.5	Critérios para avaliação da pesquisa.....	41
REFERÊNCIAS		42
ANEXOS		43
APÊNDICE		44

INTRODUÇÃO

Em 2021, em uma situação hipotética, uma pessoa que necessite descobrir a melhor rota para chegar em um local pode, por intermédio de um comando de voz, pesquisar tal rota e concluir seu objetivo de forma ágil. Ele equivale ao envio de um e-mail de suma importância que, com poucos cliques, pode abrir um conversor de fala para texto. Tais atividades só são possíveis com a evolução do reconhecimento de voz, tecnologia que faz uso de um microfone para poder captar falas e redigi-las em um texto.

Da mesma forma que o processamento de fala para texto evoluiu ao ponto de ser uma ferramenta corriqueira, o processamento de imagem também alcançou patamares nunca vistos antes, uma câmera ligada consegue fazer o reconhecimento se uma pessoa está ou não usando máscara, com a foto de um rosto é possível saber qual a emoção do indivíduo ou até mesmo identificar a placa de um carro com uma câmera instalada em uma rodovia.

Estas tecnologias também possuem suas usabilidades mais altruístas, sendo elas no auxílio durante aprendizado de crianças autistas a reconhecer expressões faciais¹, ou uma aplicação que ajuda cegos no reconhecimento de roupas², entre inúmeras outras, contudo existe uma lacuna no quesito de inclusão de deficientes auditivos. Existe uma gama considerável de ferramentas que realizam a tradução do português para a Libras (Língua Brasileira de Sinais) de uma forma muito eficiente no amparo de surdos a entender a mensagem de ouvintes que não conhecem linguagem de sinais, entretanto quando há a necessidade de uma forma de compreender uma mensagem dita em Libras isso só pode ocorrer por intermédio de um intérprete ou de uma pessoa que conheça tal dialeto. Em um país que possui uma população de mais de 10 milhões de habitantes que possuem algum tipo de deficiência auditiva segundo pesquisas realizadas em 2010 pelo IBGE³ (Instituto Brasileiro de Geografia e Estatística), que em sua maioria fazem uso e fruto da Libras como principal forma de expressão, sendo ela considerada a segunda língua oficial do país, torna-se

¹ Conforme disponível em: <<https://www.noticias.unb.br/publicacoes/117-pesquisa/119>> Acesso em: 07 de agosto de 2021.

² Conforme disponível em: <<https://www.redalyc.org/journal/3131/313158902036/html/>> Acesso em: 07 de agosto de 2021.

³ Conforme disponível em: <<https://www.ibge.gov.br/apps/snig/v1/?loc=0&cat=-1,-2,-3,128&ind=4643>> Acesso em: 07 de agosto de 2021.

inadmissível a falta de ferramentas tecnológicas que possam estreitar as barreiras do distanciamento entre deficientes auditivos e ouvintes.

Ao longo de uma pesquisa introdutória em busca de trabalhos acadêmicos relacionados ao tema de tecnologias voltadas para reconhecimento da Libras, encontram-se diversos artigos que apresentam propostas de aplicar vários tipos de ferramentas para atender tal objetivo, todavia evidencia-se que uma grande quantidade destes trabalhos depende de uma base de dados confiável e bem estruturada para que seus desenvolvimentos possam obter resultados mais satisfatórios. Inicialmente, este trabalho tinha como premissa estruturar uma base de dados própria e, com ela, desenvolver um método previsor que seguia a mesma ótica dos trabalhos acima citados, tendo como diferencial as ferramentas e metodologias aplicadas, entretanto ao ser contemplado o déficit de um conjunto de dados robusto que tenha capacidade de suportar todas as pesquisas nessa linha de estudos, optou-se em alterar a orientação deste trabalho acadêmico, chegando à pergunta orientadora, que vem à tona da seguinte forma:

Seria possível desenvolver uma base de dados bem ordenada e documentada com a capacidade de atender esta demanda?

Para responder esta pergunta, e considerando as limitações em arquitetar uma base de dados a partir de imagens, este trabalho tem como objetivo estruturar uma base de dados fazendo uso de computação visual no reconhecimento de mãos e seus pontos de referências com a finalidade de capturá-los, estruturá-los e disponibilizá-los de uma maneira para facilitar sua utilização, inicialmente focando nos seguintes sinais: Bom dia, boa tarde, boa noite, por favor, desculpa, obrigado, ajudar, entender, olá e parabéns.

Devido ao elevado número de pessoas no Brasil que possuem alguma deficiência auditiva, este trabalho poderá auxiliar indiretamente esta parcela da população que é dependente da linguagem de sinais, pois outros trabalhos poderão beneficiar-se desta pesquisa, podendo assim desenvolver novas ferramentas que possam vir a quebrar a barreira de comunicação entre os surdos e pessoas ouvintes.

Pelo fato de os autores estudarem em uma instituição de ensino superior pública, em que a preocupação social é abordada constantemente durante o curso, o desenvolvimento de um sistema para este fim vem ao encontro a missão da faculdade, além disso o desenvolvimento deste trabalho acadêmico pode-se tornar uma base para outros estudos, visto que é uma pesquisa sobre estruturação de dados a partir de processamento de imagens voltado para a área de interpretação de linguagem de sinais.

Para os autores, neste trabalho, destacam-se dois grandes pontos, primeiramente a oportunidade de construir uma base de dados que possa vir a auxiliar outros trabalhos na ampliação da inclusão de pessoas surdas na sociedade quebrando paradigmas que hoje geram um distanciamento destes indivíduos. Em segundo ponto o desafio de implementar uma nova tecnologia, ainda pouco explorada, em uma comunidade que pode obter um ganho incalculável com o sucesso deste trabalho.

Durante o desenvolvimento do trabalho, realizar-se-á uma pesquisa bibliográfica, principalmente em artigos científicos, dissertações e teses na qual os autores ainda irão fazer um levantamento de trabalhos relacionados. Após esta etapa será possível o desenvolvimento de um sistema utilizando da linguagem Python. Ao término do desenvolvimento, o sistema será posto em teste por intermédio de treinamento e avaliação de modelos preditivos fazendo uso da base de dados construída pelos autores.

Portanto, o trabalho será composto da seguinte forma: No capítulo 1, será apresentada uma breve introdução sobre a história dos surdos no Brasil e a criação das Libras, assim como um estudo linguístico de suas características e uma contextualização do conceito de computação visual. No capítulo 2, encontram-se as metodologias abordadas, explicando as variáveis propostas para a elaboração do projeto. No capítulo 3, é descrito o processo de desenvolvimento da ferramenta, e no capítulo 4 são exibidos os resultados atingidos com o projeto e algumas considerações referente ao estudo realizado.

CAPÍTULO I

Fundamentação Teórica

Conceitos chave

No embasamento desta pesquisa, optou-se por organizar este capítulo em duas partes. Primeiramente apresentam-se os conceitos chave que referenciam o trabalho, sendo eles: História dos Surdos no Brasil, Língua Brasileira de Sinais e sua fonologia, Visão Computacional e Estrutura de Dados. Na segunda parte, apresenta-se um conjunto de trabalhos relacionados a esta pesquisa, decorrentes de estudos realizados na última década.

1.1.1 História dos Surdos no Brasil

De acordo com o Strobel (2009), no ano de 1857 foi fundado no Brasil o Instituto Imperial de Surdos-Mudos pelas mãos de Dom Pedro II, em parceria com Ernest Huet, que posteriormente recebeu a alcunha de Instituto Nacional de Educação de Surdos (INES), sediado no Rio de Janeiro. Huet, também conhecido como “pai da Libras”, era um padre francês surdo responsável pela educação da língua de sinais na França. A convite de Dom Pedro II, veio ao Brasil com o propósito de desenvolver uma língua de sinais para os surdos do país. A Língua Brasileira de Sinais (Libras) tem como sua base fonológica uma mescla entre a língua de sinais francesa e características da língua de sinais utilizada pelos surdos brasileiros. O Instituto Imperial tinha como objetivo a educação de meninos surdos da alta classe social.

Em meados de 1873, é criado no Brasil, pelo Instituto de Surdos-Mudos, a iconografia dos sinais de mão da Libras para as letras do alfabeto, por autoria de Flausino José de Gama, aluno surdo do instituto. Sete anos mais tarde ocorreu o Congresso de Milão que foi uma conferência internacional de educadores de surdos, marcando a história do ensino de pessoas com deficiência auditiva ao definir a oralidade como melhor método de aprendizado em

detrimento das línguas gestuais de forma obrigatória aos países participantes, com duração aproximada a 100 anos. Tal ação foi vista com maus olhos pela cultura surda, já que endossaram a língua de sinais como forma de comunicação, o que levou a diversos movimentos contra esta decisão e ao início muitas pesquisas na área com a finalidade de legitimar a linguagem gestual.

Para Strobel (2009) este congresso teve um reflexo muito negativo na educação de surdos no Brasil. O INES recebeu uma proibição para educar os surdos no dialeto de Libras, o que acarretou uma baixa na quantidade de professores surdos nas escolas para deficientes auditivos, tendo suas vagas preenchidas por professores ouvintes. Por volta da década de 30, vários ex-estudantes do INES fundaram a primeira Associação Brasileira de Surdos-Mudos, hoje já inativa. Entre os anos de 1953 e 1956, foram fundadas outras 3 Associações de Surdos, que tiveram papéis importantes na vivência de pessoas com deficiência auditiva no país. Por intermédio destas associações, outras unidades foram fundadas em vários estados do Brasil. A Federação Nacional de Educação e Integração de Surdos (FENEIS), organização filantrópica sem fins lucrativos que luta pela inclusão dos surdos e garantia de seus direitos, teve origem no ano de 1987.

Na década de 90, a história dos surdos recebe um destaque em diversos eventos bi linguísticos (Língua de Sinais e Língua Oral) que ocorreram por todo o país, por intermédio de Universidades Federais com parceria de professores de diversos países, que ministraram cursos de metodologias de ensino de suas respectivas línguas de sinais. No final da década ocorre um congresso de extrema importância para a história da Libras: o V Congresso Latino-Americano de Educação Bilíngue para Surdos, onde uma das pautas de debate era a formação de professores surdos e intérpretes de Libras.

Como pontuou Strobel (2009) o resultado destes esforços combinados serviu como base para o início dos projetos de lei que regularizavam a Libras no Congresso, tendo seu início no ano de 1993. Foi somente no ano de 2002, mais especificamente no dia 24 de abril, que entrava em vigor a lei nº 10.436/02⁴, reconhecendo a Libras oficialmente no país, esta data ganhou grande valor para a cultura surda tornando-se reconhecida como Dia Nacional da Língua Brasileira de Sinais.

Nos treze anos seguintes várias outras leis surgiram com o objetivo de fortalecer ainda mais a lei do ano de 2002, contudo todos estes esforços realizados não foram suficientes para

⁴ Conforme disponível em: <http://www.planalto.gov.br/ccivil_03/leis/2002/110436.htm> Acesso em: 07 de agosto de 2021.

garantir uma inclusão efetiva de pessoas surdas na sociedade. Mesmo sendo considerada a segunda língua oficial do país, grande parte da população ouvinte não possui os conhecimentos desta linguagem, o que dificulta o convívio diário entre os grupos e consequentemente reforça a exclusão dos não-ouvintes da sociedade.

1.1.2 Língua Brasileira de Sinais (Libras) e sua fonologia

A definição de linguagem natural pode ser concebida como qualquer meio de comunicação desenvolvido naturalmente com a finalidade de promover a troca de informações entre seus usuários, com um conjunto limitado de componentes que permite a criação de sentenças infinitas ao uso. Tais linguagens diferem, por exemplo, das linguagens de programação de computadores, ao terem seu aprendizado por meio do convívio social dos indivíduos. Seguindo esta definição, pode-se construir um paralelo com as linguagens de sinais para com a sociedade dos surdos, visto que seus usuários possuem a capacidade de construir uma quantidade ilimitada de expressões utilizando o conjunto de funções e movimentos disponibilizados pela língua. (FONSECA, 2020)

De forma similar às demais linguagens de sinais existentes, a Libras tem sua estrutura modal baseada no princípio gestual-visual, onde faz uso de movimentos gestuais e/ou expressões faciais para transmitir uma mensagem, e a forma de recebê-la vem por intermédio da visão, o que difere da língua falada que tem como fundamentos de transmissão a fala (envio) e a audição (recepção). Entretanto, tais diferenças não se encontram única e exclusivamente em seus canais de difusão, mas também em suas estruturas gramaticais.

Fonologia⁵, do grego *phonus* (som) e *logos* (estudo), é o estudo da linguagem sonora de um idioma visando estruturar e organizar os componentes nas quais, isoladamente, não possuem significado. A Libras, por sua vez, possui suas próprias regras fonológicas devido ao fato de ser uma língua espaço-visual, sendo elas definidas em cinco vertentes principais: configuração das mãos, movimento, ponto de articulação, orientação da mão e expressão facial e/ou corporal.

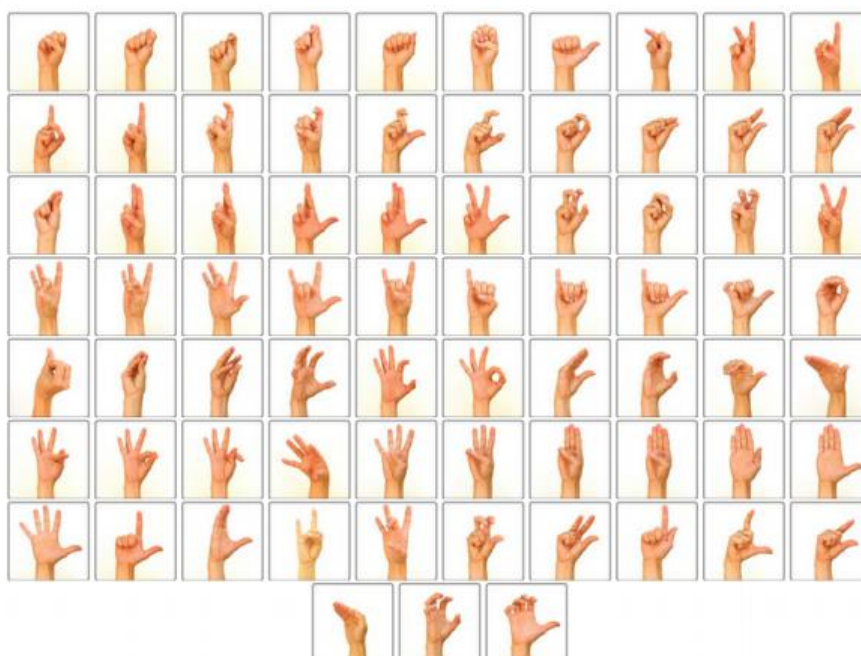
A seguir serão divididos em cinco subseções as definições citadas acima.

⁵ Conforme disponível em: <<https://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/fonologia>> Acesso em: 15 agosto de 2021.

1.1.2.1 Configuração das mãos

Consiste na forma como a mão é configurada, podendo fazer uso ou não de ambas as mãos, porém algumas delas não transmitem significados isoladamente pois necessitam dos outros parâmetros para seu entendimento, já outras assumem as formas das letras do alfabeto. Fontes de pesquisa, porém, divergem quando se trata da quantia exata de configurações existentes. De acordo com Cruz (2020), estão presentes 73 configurações de mãos na Libras, conforme a Figura 1. Para Tanya e Myrna (2007, p. 28), existem 64 parâmetros, como demonstrado na Figura 2.

Figura 1: Configurações de mão extraídas do Dicionário Digital da INES



Fonte: Cruz, 2020, p.12 apud Lira e Souza, 2018

Figura 2: Configuração de Mão da Libras

Fonte: Felipe e Monteiro, 2007, p.28

1.1.2.2 Movimento

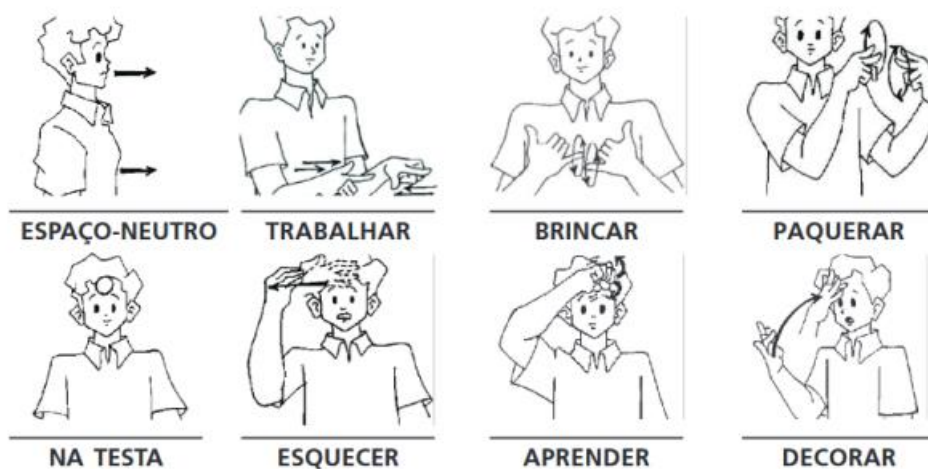
No parâmetro de movimento, sinais podem ser subdivididos em dois grupos: sinais estáticos, aqueles que não fazem uso de movimentos das mãos para serem executados, e sinais dinâmicos, que necessitam da movimentação das mãos. A forma da movimentação das mãos não implica só na execução dos sinais, mas também na intensidade na qual são expressados, no modo adverbial ou em sua temporalidade (CRUZ, 2020). Exemplos de sinais que utilizam movimento estão dispostos na Figura 3.

Figura 3: Representação do parâmetro “movimento”

Fonte: Cruz, 2020, p.14 apud Felipe e Monteiro, 2007

1.1.2.3 Pontos de articulação

A utilização dos pontos de articulação baseia-se na predominância do local onde o sinal de mão é realizado, podendo ser estes em áreas neutras (região em frente ao tronco), ou que faz uso de alguma parte do corpo para sua realização, como por exemplo a área da cabeça/rosto, ou a área dos membros. Um exemplo que diferencia sinais com a mesma configuração de mão baseado no ponto de articulação encontra-se na Figura 4.

Figura 4: Representação do parâmetro “ponto de articulação”

Fonte: Cruz, 2020, p.13 apud Felipe e Monteiro, 2007

1.1.2.4 Orientação da mão

Expressa o sentido na qual a mão segue em casos de sinais que requerem movimentos direcionáveis. Tais orientações podem expressar oposição de sentido, mesmo com a realização de movimentos e configuração de mãos similares. Exemplos disponíveis na Figura 5.

Figura 5: Representação do parâmetro “orientação”

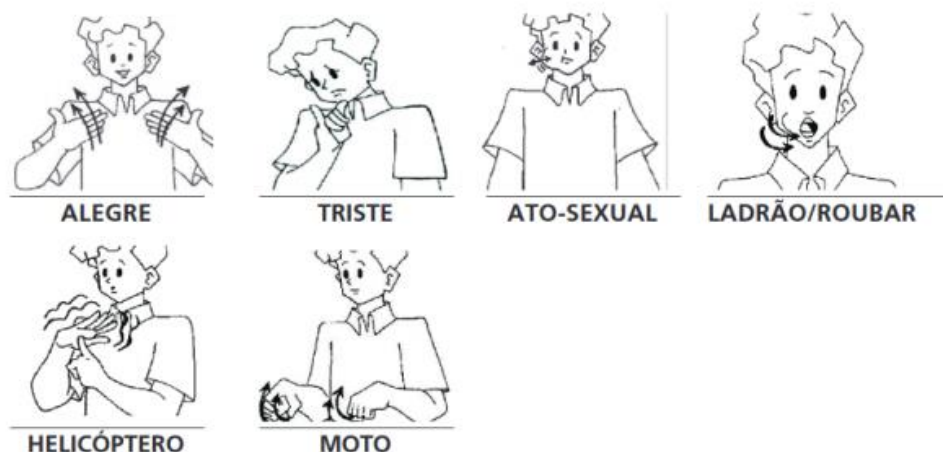


Fonte: Cruz, 2020, p.14 apud Felipe e Monteiro, 2007

1.1.2.5 Expressões

Nos parâmetros citados acima, há a predominância do uso das mãos para sua execução, contudo alguns sinais também podem fazer uso de expressões faciais e/ou corporais para expor significado ou intensidade. Na Libras, porém, existem sinais / expressões que não utilizam das mãos para sua realização, dependendo de expressões para sua transmissão.

Figura 6: Representação do parâmetro “expressão”



Fonte: Cruz, 2020, p.15 apud Felipe e Monteiro, 2007

1.1.3 Processamento de Imagens e Visão computacional

Distinguir a diferença entre processamento de imagens e visão computacional é uma tarefa extremamente árdua. Pode-se definir processamento de imagens como a entrada de uma imagem em um sistema que retorna um conjunto de valores, sejam eles uma outra imagem ou não. Já a visão computacional tenta emular a visão humana, onde o sistema também tem uma imagem como entrada, contudo sua saída será na forma de uma interpretação, parcial ou não desta imagem.

1.1.3.1 Processamento de Imagens

Em grande suma, um sistema de visão computacional necessita inicialmente uma etapa de processamento de imagens, em alguns casos a imagem necessite de um pré-processamento sendo ele um redimensionamento da imagem, uma conversão de formato ou um filtro com a finalidade de remover ruídos que possam impactar na real compreensão da imagem.

Para Marengoni e Stringhini (2008) estes ruídos podem ter fontes causadoras bem distintas, seja ela a ferramenta utilizada para a captura da imagem, a iluminação do ambiente, localização da câmera correlacionada ao objeto de captura, intempéries climáticas, entre outras. Para tratar esses ruídos em grande parte dos casos são aplicados filtros corretivos nas imagens obtidas para melhorar sua análise.

1.1.3.2 Visão computacional

É sabido que o ser humano faz uso da visão para capturar dados ao seu redor e realizar seu processamento, obtendo assim, uma melhor compreensão do mundo que o rodeia. Pode-se citar como exemplo, quando uma pessoa olha uma árvore ele pode apresentar suas características tais como cor, tamanho, tipo de folhagem, se é frutífera entre outras. Assim, como os humanos, os animais podem fazer uso da visão para obter alimentos ou perceber situações de perigo.

Já a visão computacional responde por agilizar a interpretação de imagens e vídeos de um método processual sem precedentes (VELUDO, 2020). Esta visão possui um alto grau de complexidade visto que busca mimetizar a visão e compreensão dos seres biológicos. Quando comparado a um ser humano que consegue interpretar padrões de formas, cores ou sombras sem um menor esforço, os algoritmos de aprendizado de máquinas são propelidos ao erro, minimizando desta forma sua eficácia.

1.1.4 Estrutura de dados

O conceito de estruturação de dados teve sua origem na necessidade gerada pelo avanço da tecnologia em armazenar informações de forma eficiente. As estruturas de dados (EDs) permitem ser modificadas ou lidas, gerando novos ramos ou versões de uma ED a cada edição em seu conjunto (COUTO, 2018).

Comumente, as estruturas de dados são formadas por elementos indexados com uma chave, e por meio desta é executada diversas funções para sua localização e modificação, como por exemplo (CORMEN, 2009):

- SEARCH(S, k): Uma consulta que, ao receber uma estrutura S e uma chave k, retornará um ponteiro para um elemento correspondente, ou nulo caso nenhum seja encontrado;
- INSERT(S, x): Uma operação de modificação que incrementa uma estrutura S com o elemento x;
- DELETE(S, x): Uma operação de modificação que, dado um valor x de um elemento em uma estrutura S, o removerá do conjunto;
- MINIMUM(S): Uma consulta que retorna um ponteiro para o elemento com a menor chave de uma estrutura S;

- MAXIMUM(S): Uma consulta que retorna um ponteiro para o elemento com a maior chave de uma estrutura S;
- SUCESSOR(S, x): Uma consulta que, dado um elemento x com sua chave vinda de um conjunto S ordenado, retorna um ponteiro para a próxima maior chave de elemento da estrutura, ou nulo caso x seja o maior elemento;
- PREDECESSOR(S, x): Uma consulta que, dado um elemento x com sua chave vinda de um conjunto S ordenado, retorna um ponteiro para a próxima menor chave de elemento da estrutura, ou nulo caso x seja o menor elemento;

Em estrutura de dados, um tipo de dado é definido como um conjunto de valores que podem ser atribuídos a variáveis executadas por um programa, assim como os processos que podem ser aplicados a ele. Esta tipificação pode assumir duas formas, a de dados básicos ou primitivos e dados estruturados (EDELWEISS; GALANTE, 2009).

Tipo de dados básicos recebem essa denominação por não possuírem uma estrutura sobre seus valores, sendo eles a menor decomposição que podem vir a assumir, as principais linguagens de programação classificam seus dados primitivos como:

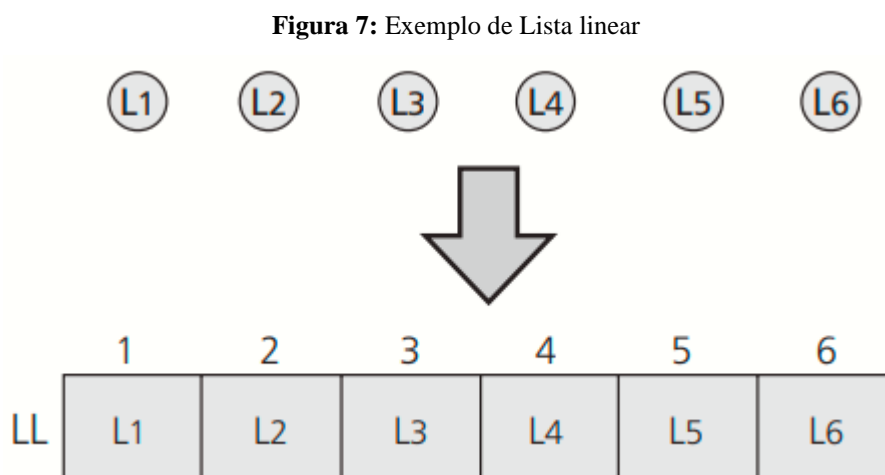
- Lógicos, podendo assumir valores verdadeiro ou falso;
- Inteiros, para valores inteiros;
- Reais, abrangendo valores fracionais;
- Caractere, representando os valores de caracteres.

Conforme apresentado por Rissetti (2009) em seu livro, dados estruturados poder ser caracterizados como:

- Listas: Coleção de elementos organizados de forma linear não sequencial;
- Filas: Coleção de elementos organizados sequencialmente, onde o primeiro a entrar deve ser o primeiro a sair;
- Pilhas: Coleção de elementos organizados sequencialmente, onde o último a entrar deve ser o primeiro a sair;
- Árvores: Coleção de elementos não-linear, de forma bidimensional, utilizada para implementação de hierarquias.

1.1.4.1 Listas

O método mais usual de se implementar uma lista pode ser feito por uso de arranjos unidimensionais (vetores), onde cada elemento da lista recebe a denominação de nodo. Para acessar um nodo da lista basta fazer uso de seu índice no arranjo, responsável por representar sua localização na lista. A Figura 7 apresenta uma lista constituída de 6 nodos, identificado pelo arranjo LL com 6 elementos (EDELWEISS; GALANTE, 2009).

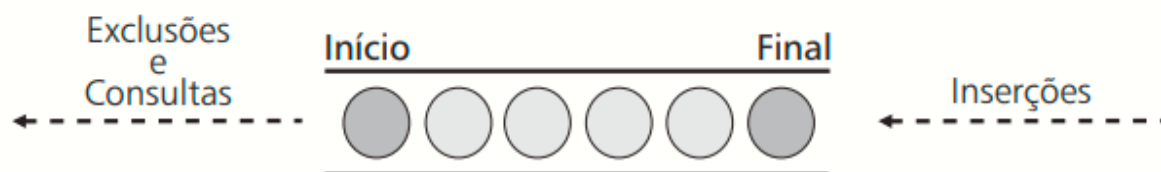


Fonte: Edelweiss e Galante, 2009, p.53

Todos os elementos formadores do arranjo precisam pertencer ao mesmo tipo primitivo de dados, responsáveis por definir os nodos da lista. Permitir acessar qualquer nodo da lista por intermédio de seu índice garante a lista uma característica muito importante, visto que não há a necessidade de percorrer toda a lista para poder acessar um nodo específico.

1.1.4.2 Filas

A fila pode ser definida como uma espécie de lista que permite o acesso única e exclusivamente de seus elementos extremos. Incluir novos nodos a uma fila devem sempre serem feitos ao final da fila, assim como consultar, alterar ou remover um nodo da lista pode somente ser executado a partir do primeiro nodo da fila (Figura 8). Esta característica é denominada como FIFO (First In First Out) também conhecido em português como PEPS (Primeiro que Entra, Primeiro que Sai).

Figura 8: Exemplo de Fila

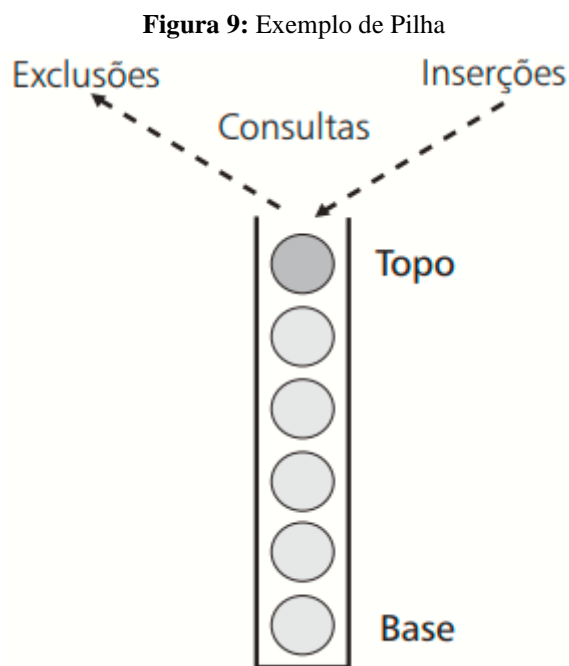
Fonte: Edelweiss e Galante, 2009, p.137

Um dos principais exemplos da usabilidade de filas pode ser citado como o gerenciamento de impressões de multiusuários, onde somente uma requisição pode ser executada por vez, sendo a primeira a ser executada é justamente a que foi enviada primeiramente (EDELWEISS; GALANTE, 2009).

1.1.4.3 Pilhas

Assim como a fila, uma pilha pode ser definida como uma lista com propriedades exclusivas, a manipulação de seus nodos acontece somente em uma de suas extremidades, denominada de topo da pilha. Sejam uma consulta, alteração, inclusão ou remoção, devem ser feitos somente sobre um nodo sendo ele o que ocupa o lugar do topo da pilha (Figura 9). Esta característica recebe o nome de LIFO (Last In, First Out), também conhecida como UEPS (Última que Entrar, Primeiro que Sair).

Pilhas são amplamente utilizadas quando o sistema precisa de um controle de histórico, por exemplo a navegação entre páginas da web, onde quando se faz uso da funcionalidade de regressar no navegador, o sistema remove a página da pilha permitindo o acesso à página anteriormente adicionada. Editores de texto também largamente fazem uso de pilhas para poderem armazenar as alterações de um texto, permitindo assim que o usuário possa regressar suas alterações removendo elementos da pilha de alterações criada (EDELWEISS; GALANTE, 2009).



Fonte: Edelweiss e Galante, 2009, p.127

1.1.4.4 Árvores

Árvores apresentam o maior grau de complexidade dentre as estruturas de dados por apresentar diversas usabilidades e classificações. Em suma uma árvore pode ser caracterizada por uma estrutura hierárquica onde um nodo da estrutura se relaciona com um ou mais nodos de maneira subordinada. A Figura 10 ilustra de maneira clara a estrutura hierárquica de uma árvore (EDELWEISS; GALANTE, 2009).

Estas estruturas hierárquicas podem assumir muitos significados, sejam eles:

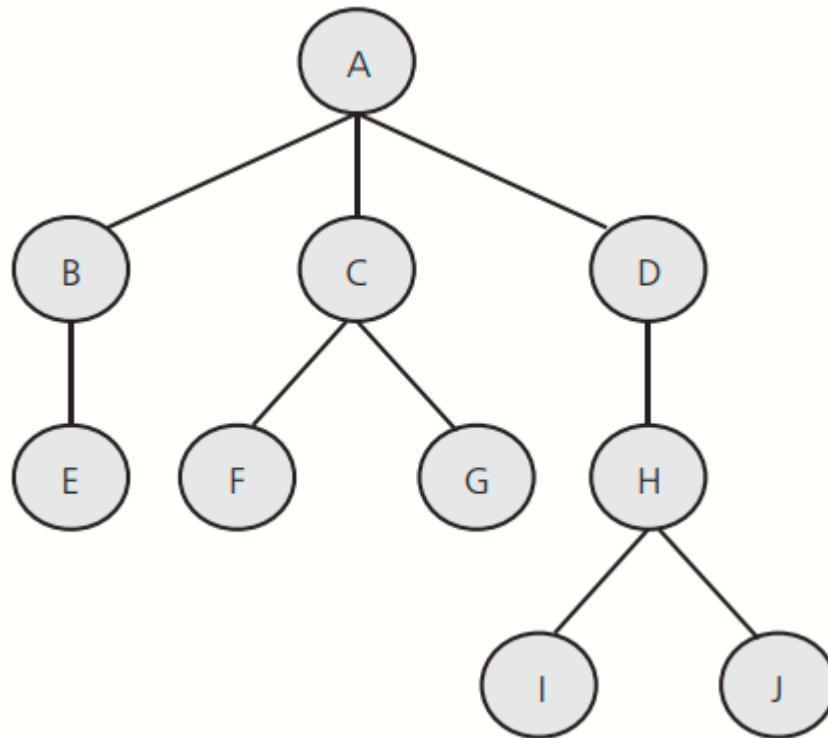
- Especialização, representando classes e subclasses, onde cada classe pode ser especializada em outras categorias (Figura 11);
- Composição, uma classe é composta de outras subclasses que representam suas propriedades ou características fundamentais (Figura 12);
- Dependência, onde uma classe é responsável por suportar suas subclasses (figura 13).

A estrutura que compõem uma árvore pode receber várias nomenclaturas dependendo da publicação, contudo há três propriedades primordiais para sua configuração:

- Raiz: Nodo principal da árvore, a partir dele todos os nodos tornam-se seus subordinados;

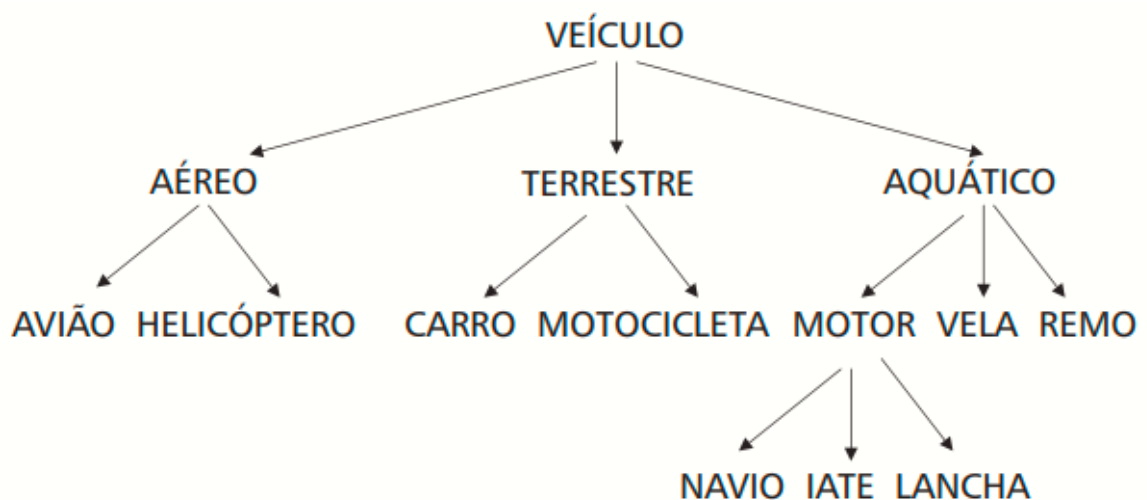
- Nodos descendentes: Recebem esta denominação por tratarem os nodos que se relacionam em graus de parentescos, onde um nodo pode vir a ser filho de outro nodo que receberia a denominação de pai;
- Folhas: São os nodos que não apresentam descendência abaixo deles.

Figura 10: Exemplo de Árvore

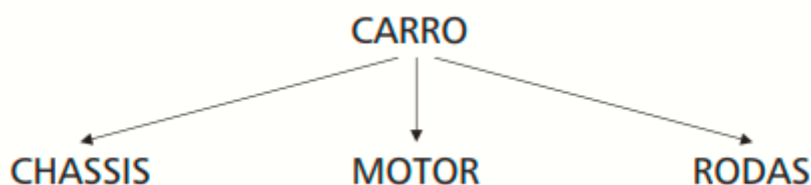


Fonte: Edelweiss e Galante, 2009, p.168

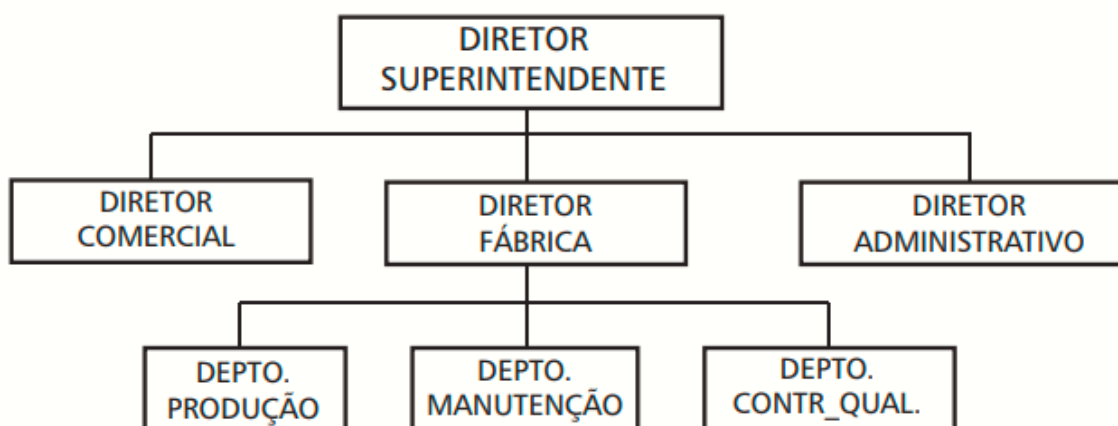
Figura 11: Exemplo de Árvore de Especialização



Fonte: Edelweiss e Galante, 2009, p.169

Figura 12: Exemplo de Árvore de Composição

Fonte: Edelweiss e Galante, 2009, p.169

Figura 13: Exemplo de Árvore de Dependência

Fonte: Edelweiss e Galante, 2009, p.169

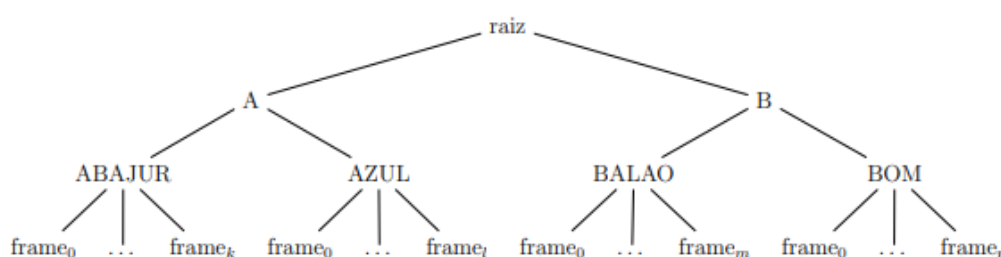
1.2 Trabalhos relacionados

Esta seção tem como função exemplificar resumidamente os trabalhos utilizados para a estruturação deste estudo apresentado. Como fonte de pesquisa foi utilizado o Google Scholar para a obtenção destes trabalhos. Tal levantamento baseou-se principalmente em trabalhos acadêmicos e artigos científicos correlacionados à implementação de ferramentas para compreensão de linguagem de sinais ou que fizessem uso de tais ferramentas em outras áreas de aplicação entre os anos de 2000 e 2021.

Fonseca (2020) criou uma base de dados mais bem estruturada para utilizar no desenvolvimento de um classificador de imagens relacionadas à configuração de mãos da Libras e podendo ser utilizada em outros trabalhos relacionados. Como metodologia, Fonseca utilizou ferramentas de 'web-crawler' para extrair do dicionário online APILRJ de Língua Brasileira de Sinais os dados necessários para seu estudo. Após a obtenção dos dados, Fonseca estruturou os dados obtidos em cinco categorias, sendo elas: letra, palavra, classe gramatical, configuração de mão e assunto. Para seu estudo estatístico durante a revisão dos

dados, Fonseca optou pela remoção de dados discrepantes ou imprecisos, buscando desta forma uma base de dados mais confiável. Com o intuito de evitar o enviesamento de seu modelo predictor, Fonseca realizou uma análise estatística de sua base de dados, identificando padrões e comportamentos dos dados coletados. Ao término desta análise, a autora pôde obter um melhor esclarecimento do comportamento de seu conjunto de dados. Para o processamento das imagens no modelo predictor, Fonseca utilizou de redes neurais artificiais e suas funções, como '*backpropagation*' e função de custo, entre outras. Todo este processamento teve sua base primordialmente em Python, fazendo uso principalmente das bibliotecas de ciências de dados, sendo elas Tensorflow, Sklearn, Numpy e Pandas. Para testar os resultados de seu modelo predictor Fonseca utilizou, além de sua própria base de dados, uma base de imagens da Língua de Sinais Americana (ASL), dividindo-as em uma proporção de 80% para treinamento e 20% para teste. Em seus experimentos, Fonseca conclui que em seu estudo foram obtidos resultados satisfatórios na identificação de configurações de mãos em imagens de Linguagem de Sinais Americana (ASL), contudo quando sua arquitetura era exposta à base de dados da linguagem de sinais nacional produzida, a classificação das configurações de mãos era imprecisa, levando-a a identificar que o padrão utilizado nas imagens não era ideal para seu método.

Figura 14: Exemplo de parte da estrutura do diretório da base de dados

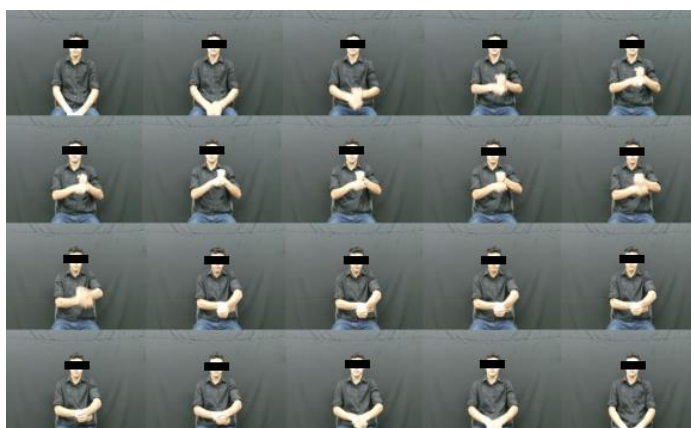


Fonte: Fonseca, 2020, p.19

Cruz (2020) apresentou uma estratégia de reconhecimento de gestos estáticos e dinâmicos da Língua Brasileira de Sinais utilizando dados das dimensões temporais e espaciais em um modelo de redes neurais profundas. Após pesquisas de trabalhos correlatos e estudos de avanços das tecnologias da área, Cruz selecionou os modelos e técnicas a serem utilizadas em seu trabalho, optando pelo modelo 3D-CNN de redes neurais e de dados RGB e fluxo óptico. Finalizando a configuração dos parâmetros em seu modelo predictor, Cruz fez uso de uma base de dados com cerca de 23 mil vídeos de 2 a 5 segundos cada para treinar, validar e testar seu método. Com isso, uma das premissas de seu trabalho era implementar as

técnicas de aumento e fusão de dados, além da transferência de aprendizado na base de vídeos inicial. Para o aumento de dados, Cruz fez uso da metodologia tradicional de combinação de imagens e modificação de cores. Foram implementadas as estratégias de fusão antecipada e tardia nos dados RGB e de fluxo ótico, abordagem multicanal da fusão de dados. Por fim, também implementou a transferência de aprendizado de modelos pré-treinados de reconhecimento de ações adaptando-os para seu objetivo de reconhecimento da Libras. Como sua base de dados era constituída de vídeos com sete pessoas diferentes realizando os mesmos sinais pré-estipulados, Cruz optou por segregar os dados de um destes membros para utilizá-los em sua fase de testes. Cruz conclui que, devido às técnicas aplicadas para aumento de dados, seu trabalho destaca-se pela base de dados criada para implementação do método previsor e mostra que os resultados obtidos foram satisfatórios e demonstram avanço quando comparados aos trabalhos relacionados.

Figura 15: Exemplo de instância da base de dados

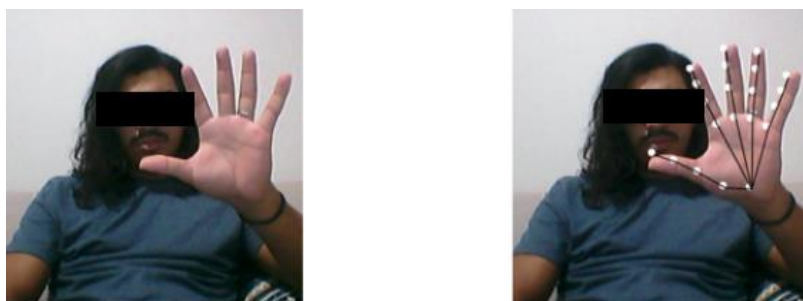


Fonte: Adaptada de Cruz, 2020, p.46

Nascimento et al. (2020) compilaram informações capazes de desenvolver um minicurso introdutório visando exemplificar a biblioteca Tensorflow, assim como modelos aptos a reconhecer em tempo real pontos de articulação responsáveis por representar a postura humana com a finalidade de auxiliar o desenvolvimento de novas ferramentas aplicadas à saúde postural. Para tal tarefa foi feito uso da linguagem de programação Python para estruturar os códigos apresentados no trabalho, baseando-se principalmente nas bibliotecas Tensorflow e MediaPipe com funções de elucidar os modelos de reconhecimento de postura e mapeamento de pontos de articulação respectivamente. Como medida de aplicabilidade deste trabalho, Nascimento et al. fizeram uso de dois trabalhos desenvolvidos por membros do Laboratório de Neuro-Inovação Tecnológica & Mapeamento Cerebral (NitLab) da

Universidade Federal do Delta do Parnaíba (UFDPAr), uma ferramenta mobile avaliativa de postura capaz de identificar e pontos anatômicos conectando-os, podendo assim determinar segmentações corporais responsáveis por desvios posturais, a outra aplicação usada é um jogo desenvolvido também pelo NitLab em plataforma web com o objetivo de reabilitação neuro funcional de membros superiores para pacientes que tenham sofrido AVC. Nascimento et al. puderam, com este trabalho, pontuar de maneira clara e objetiva a participação recorrente da inteligência artificial aplicada à saúde, seja ela por meio de prevenção de problemas posturais, seja ela com a finalidade de desenvolver tratamentos para problemas de alta complexidade.

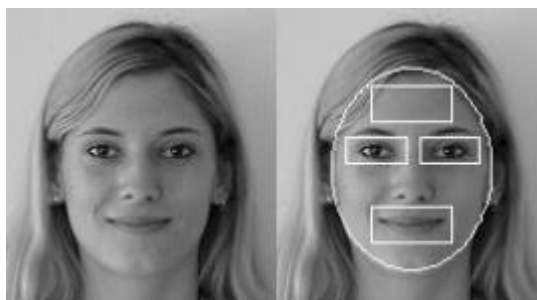
Figura 16: Mapeamento da mão em tempo real usando o HandPose



Fonte: Adaptada de Nascimento et al., 2020, p.138

Marengoni e Stringhini (2009) realizaram uma pesquisa focada na usabilidade teórico-prática da biblioteca OpenCV (Open Source Computer Vision) a fim de apresentar de maneira subjetiva um tutorial sobre processamento de imagem e visão computacional. Neste tutorial apresentado os autores buscam fundamentar seu leitor de maneira simples e didática todos os passos necessários para pôr em prática uma aplicação baseada em OpenCV, desde sua instalação até a implementação de modelos para processamentos e compreensão de imagens. Destaca-se neste trabalho a metodologia de apresentar o conteúdo, sendo ele dividido em conceitos fundamentalistas e de implementação prática, abrangendo desta forma todo o espectro de aprendizado. Ao término do trabalho, Marengoni e Stringhini puderam expor de uma forma robusta e concreta uma visão completa das funcionalidades da ferramenta OpenCV, assim como puderam delimitar as fronteiras que dividem o entendimento dos conceitos de processamento de imagem e visão computacional.

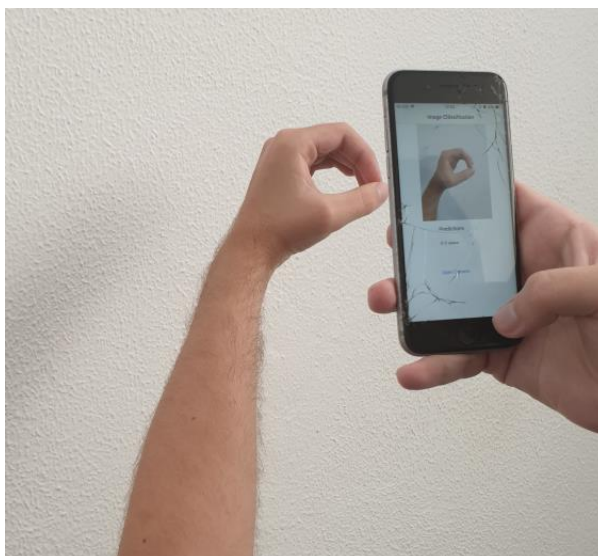
Figura 17: Exemplo de estrutura para o reconhecimento de faces



Fonte: Marengoni e Stringhini, 2009, p.144

Nogueira (2020) buscou implementar uma aplicação que fosse capaz de transcrever em palavras sinais gesto-facial da Linguagem de Sinais Portuguesa com a finalidade de auxiliar na compreensão dela para ouvintes que não possuam qualquer conhecimento na linguagem. Para tal objetivo, foi implementado métodos de inteligência artificial focados em Deep Learning. O modelo usado para o treinamento foi o Mobile-Net v2, encontrado na biblioteca PyTorch. Durante o processo avaliativo, Nogueira dividiu-o em duas frentes, sendo elas uma base estatística gerada durante o processo de treinamento e testes de captura de imagem por intermédio de uma câmera de celular para serem processadas pelo modelo previamente treinado. Nogueira aponta com este trabalho que, mesmo seu modelo tendo atingido valores próximos ao 99% de acuracidade durante o treinamento, quando posto em prática, não atendia às expectativas esperadas, tendo em vista a quebra da barreira de comunicação entre deficientes auditivos que fazem uso de linguagem de sinais e ouvintes que usam da língua oral para comunicar-se.

Figura 18: Teste do sistema SignPic em iOS

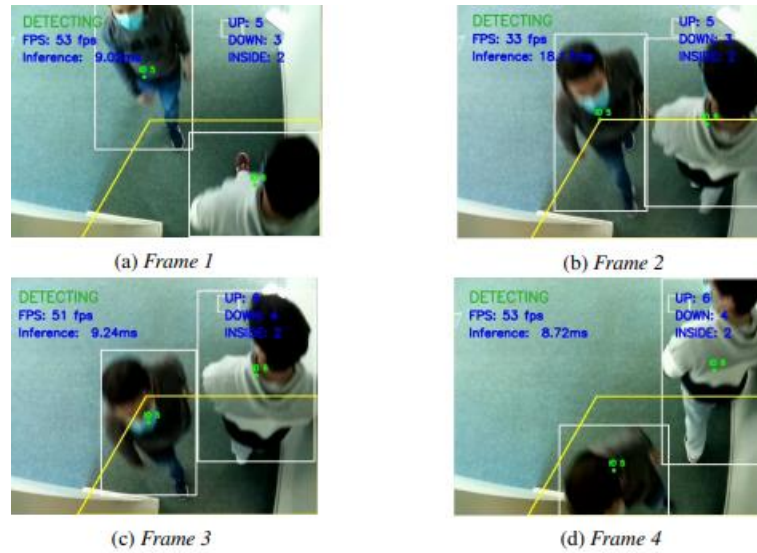


Fonte: Nogueira, 2020, p.49

Martins (2021) manifesta a problemática enfrentada durante o período de pandemia do Covid-19 onde há uma necessidade de controle populacional em ambientes internos a fim de evitar grandes aglomerações, para a solução desta situação o autor desenvolveu um sistema de contagem de pessoas inteligente com alta precisão que pode funcionar em tempo real. Martins dividiu sua aplicação em três frentes: a criação do algoritmo responsável pela contagem de pessoas, a escolha do hardware que iria auxiliar o algoritmo a desenvolver sua atividade e a criação de um quadro informativo com objetivo de apresentar a análise estatística das informações obtidas pelo algoritmo em tempo real. Para o desenvolvimento do algoritmo de contagem de pessoas, foi implementado um modelo de processamento de imagem chamado MobileNetv2-SSD, um detector baseado em Deep Learning. Na premissa de hardware, o algoritmo desenvolvido foi implementado em um Raspberry Pie, conectado a uma câmera de lente grande angular para a captura de imagem. Por fim a etapa de desenvolvimento do quadro informativo realizou-se fazendo uso do software ThingsBoard, que teve a implantação também de um alarme que soa ao atingir a lotação máxima do local. O processo de teste teve duas etapas de aplicação, sendo um deles realizado em ambiente de desenvolvimento, onde o sistema fez uso de um dataset de 150 imagens para seu treinamento e 50 imagens para teste, onde obteve uma acuracidade de 99% na análise das imagens de teste. Na etapa de avaliação em ambiente de produção, a aplicação foi instalada na biblioteca da Faculdade de Engenharia da Universidade do Porto (FEUP), tendo resultados acima dos 90% de acuracidade em um dia de teste. Ao término do trabalho, o autor pode introduzir uma ferramenta capaz de auxiliar,

aqueles que fizeram uso dela, na detecção de lotação de ambientes fechados a fim de evitar grandes aglomerações de maneira precisa e dinâmica.

Figura 19: Sistema para detecção e contagem de pessoas



Fonte: Martins, 2021, p.80

Quadro 1: Trabalhos relacionados

Autores	Tópicos Correlatos	Pontos Positivos	Pontos Negativos
Fonseca (2020)	Criação de Base de Dados; Libras; Tensorflow; Python;	Atingiu resultado satisfatório para ASL; Criou uma base de dados robusta para o estudo; Disponibilizou os códigos-fonte do projeto;	Não atingiu resultado satisfatório para Libras com sua base de dados;
Cruz (2020)	Criação de Base de Dados; Libras; Python;	Implementou efetivamente técnicas para criar uma base de dados bem estruturada;	Encontrou limitações pois dados originais utilizados possuíam fundo estático, atrapalhando o aprendizado do modelo;
Nascimento et al. (2020)	MediaPipe; Tensorflow; Python;	Aplicaram a inteligência artificial na área da saúde; Realizaram a leitura de pontos de referência também para o corpo;	Autores não relataram pontos de dificuldade ou melhorias.
Marengoni e Stringhini (2009)	OpenCV;	Desenvolveram uma documentação robusta de utilização da ferramenta;	Trabalho não se enquadra como pesquisa científica.
Nogueira (2020)	Libras; OpenCV; MediaPipe; Tensorflow; Python;	Atingiu 99% de acuracidade com seu método; Aplicou a captura de imagens utilizando camera de celular;	Sistema desenvolvido somente para iOS; Identificou somente sinais estáticos;
Martins (2021)	Criação de Base de Dados; Python;	Atingiu acuracidade acima de 90%; Auxiliou no controle de lotação de ambientes durante a pandemia; Disponibilizou e detalhou trechos do código-fonte do projeto;	Autores não relataram pontos de dificuldade ou melhorias.

Fonte: Autoria própria

CAPÍTULO II

Metodologia

2.1 Natureza da Pesquisa

Para o método de pesquisa, optou-se pelo método experimental, visto que Gil (1991) define a experimentação como a escolha de um objeto a ser estudado que possa vir a sofrer influências de variáveis selecionáveis, a definição das formas como controlar e, por fim, a observação dos efeitos produzidos pelas variáveis selecionadas.

2.2 Variáveis de análise

Na arquitetura deste trabalho, foram supracitadas as seguintes variáveis durante este desenvolvimento:

- Atributo dos pontos de articulação
- Características da amostra
- Dimensionamento do tempo de captura dos pontos de articulação
- Estrutura do banco de dados

2.3 Definição e apresentação das Ferramentas para Análise

As tecnologias e ferramentas utilizadas para o desenvolvimento deste trabalho são descritas a seguir:

2.3.1 Python

Python⁶ é uma linguagem de programação de alto nível, orientada a objetos, de fraca tipagem, desenvolvido por Guido von Rossum em 1991. Seu desenvolvimento e manutenção é mantido pela organização Python Software Foundation, possui uma clara legibilidade de seu código, estruturada em uma biblioteca básica poderosa, tendo também um acervo amplo de módulos e frameworks desenvolvidos por terceiros.

Graças a suas características, é fortemente usada para processamentos de dados como textos, áudios ou imagens, assim como em pesquisas científicas. Seu nome tem como origem o grupo humorístico britânico Monty Python Fly Circus.

2.3.2 OpenCV

O OpenCV⁷ (Open Source Computer Vision), desenvolvido pela Intel Corporation, é uma biblioteca 'open source' de programação que implementa diversas ferramentas de apoio à interpretação de imagens de forma computacional (MARENGONI e STRINGHINI, 2009). Com esta ferramenta, é possível realizar o processamento de imagens em vários níveis, desde a aplicação de filtros simples até funções mais complexas de reconhecimento e análise da captura, visando a acessibilidade de programadores a tecnologias de robótica e interações em tempo real humano-máquina.

Segundo Marengoni e Stringhini (2009, p. 126, apud Gonzalez et al., 2006) pode-se dividir em três níveis (baixo-nível, nível-médio e alto-nível) o espectro entre processamento de imagem e a visão computacional:

Os processos de baixo-nível envolvem operações primitivas, tais como a redução de ruído ou melhoria no contraste de uma imagem. Os processos de nível-médio são operações do tipo segmentação (particionamento da imagem em regiões) ou classificação (reconhecimento dos objetos na imagem). Os processos de alto nível estão relacionados com as tarefas de cognição associadas com a visão humana.

⁶ Conforme disponível em: <<https://www.python.org/doc/>> Acesso em: 30 de outubro de 2021.

⁷ Conforme disponível em: <<http://sourceforge.net/projects/opencvlibrary>> Acesso em: 07 de setembro de 2021.

2.3.3 MediaPipe

O MediaPipe⁸ é um framework de código aberto para desenvolvimento de modelos de aprendizado de máquina voltados para processamento de imagens, este foi desenvolvido e vem sendo mantido pela Google. O MediaPipe permite que o usuário tenha acesso a modelos previamente treinados que possuem fácil implementação buscando sempre auxiliar na pesquisa de novas funcionalidades. Possui suporte para as plataformas iOS, Android, Windows, Linux e Web.

Sua base de modelos pode ser subdividida em reconhecimento faciais, rastreamento de mãos, pose corporal, reconhecimento de objetos entre outros. Destaca-se o rastreamento de mãos, para seu desenvolvimento foram implementados 21 pontos de articulação para cada mão, apresentados na Figura 7. (Nascimento et al., 2020)

Figura 20: Pontos de articulação da mão



Fonte: Documentação MediaPipe (2020)

2.4 Experimento de Pesquisa

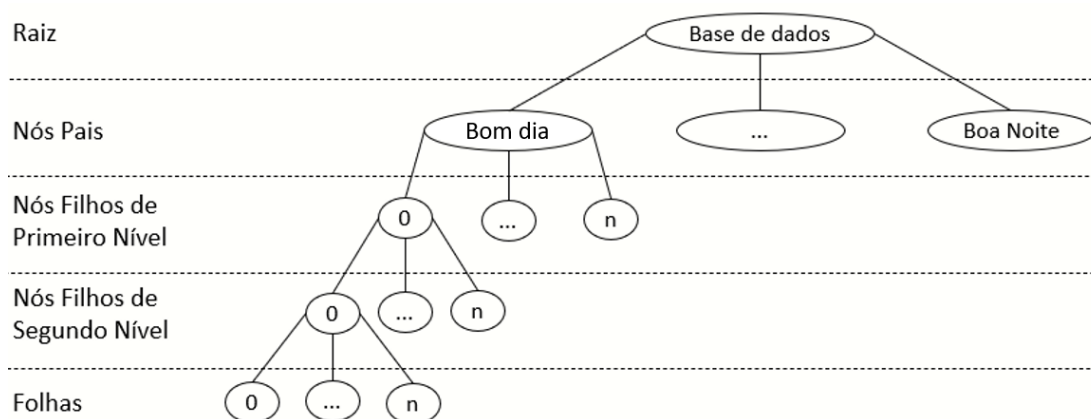
Esta experimentação busca implementar um algoritmo capaz de estruturar uma base de dados ordenada e bem classificada de pontos de referência capturados de imagens fornecidas da linguagem brasileira de sinais. Para o estudo foram analisados os seguintes sinais: Bom dia, boa tarde, boa noite, por favor, desculpa, obrigado, ajudar, entender, olá e parabéns.

⁸ Conforme disponível em: <<https://google.github.io/mediapipe/solutions/hands>> Acesso em: 30 de outubro de 2021.

Foi definido uma padronização no armazenamento dos dados obtidos durante a captura das imagens para fins de facilmente serem classificados e acessados, estruturando uma base de dados seguindo os paradigmas da Estrutura de Dados apresentados na seção 1.1.4 deste trabalho.

Inicialmente optou-se por estruturar os dados pelo modelo de árvore , onde a raiz desta árvore será a base de dados, seus nós-pais serão os sinais a serem obtidos, os nós filhos se subdividem em duas etapas, os nós filhos de primeiro nível são responsáveis pela quantidade de blocos de frames que serão armazenados, abaixo deles vem os nós filhos de segundo nível com a responsabilidade de guardar os frames individualmente, por fim temos os nós folhas, sendo eles o dado bruto, na forma de coordenada X, Y e Z no plano Cartesianos.

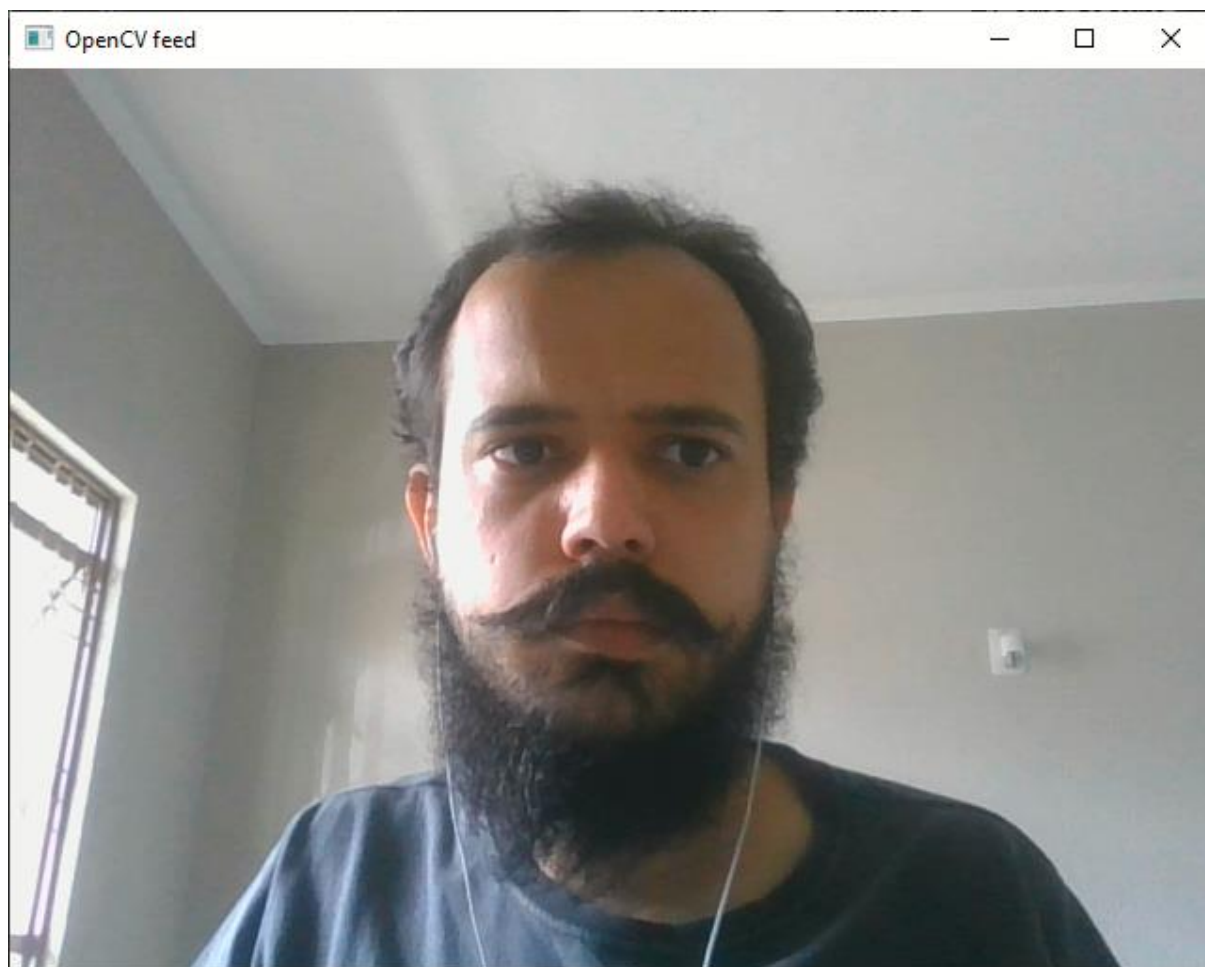
Figura 21: Árvore da base de dados



Fonte: Autoria própria

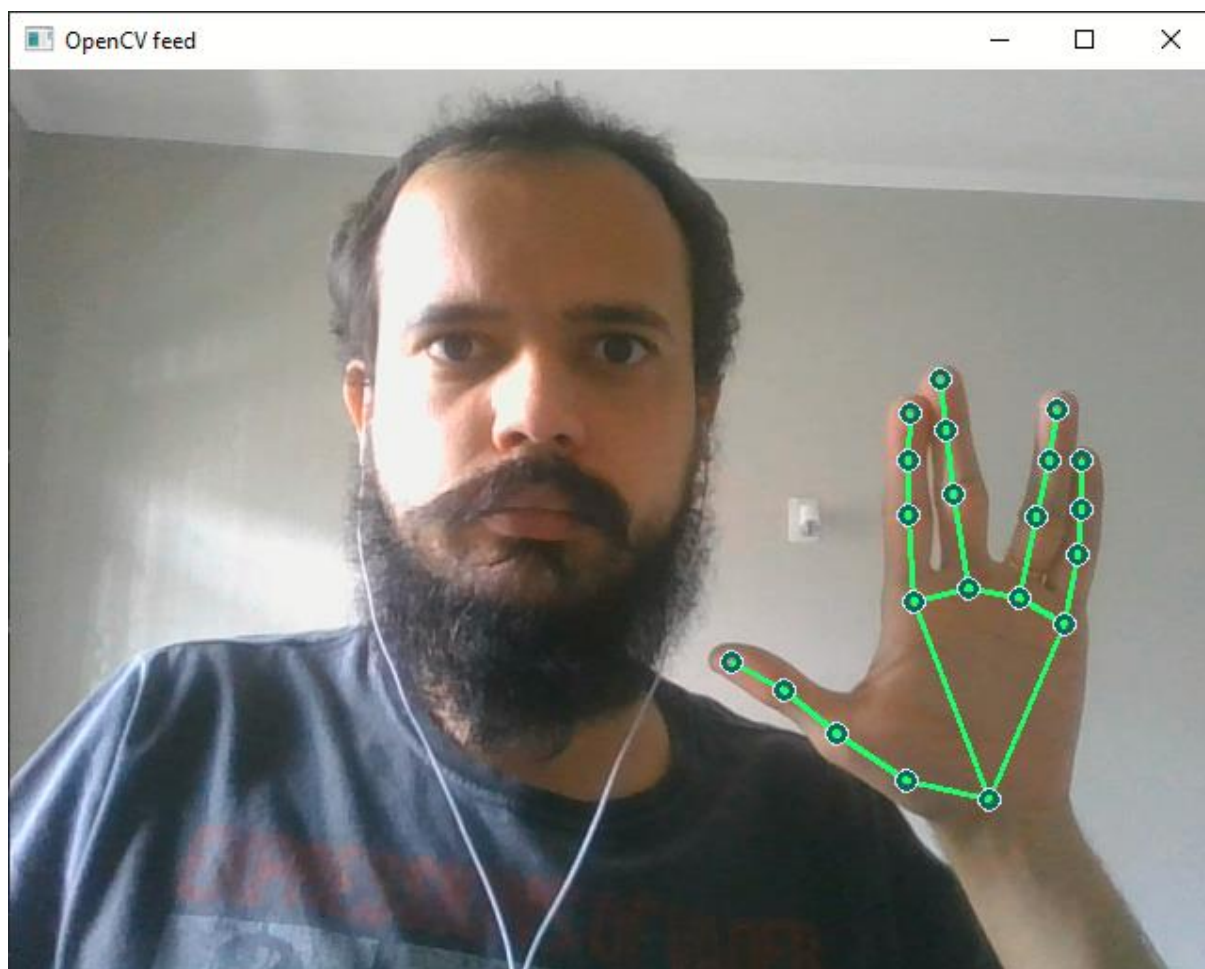
Com a estrutura de dados definida o passo seguinte é desenvolver o algoritmo responsável pela captura das imagens, neste momento faz-se uso do framework OpenCV para mapear e acessar a câmera, assim como realizar a visualização da captura em tempo real a fim de confirmar as funcionalidades que serão empregadas posteriormente.

Figura 22: OpenCV acessando câmera



Fonte: Autoria própria

Trabalhando de maneira simbiótica com o OpenCV, o MediaPipe tem sua responsabilidade em realizar o desenho dos pontos de referência na imagem de retorno do OpenCV, da mesma forma que constrói as conectividades entre os pontos, demonstrando de maneira concisa os dados a serem capturados e suas correlações.

Figura 23: MediaPipe mapeando pontos de referência

Fonte: Autoria própria

OpenCV acessando corretamente a câmera, MediaPipe mapeando os pontos de referência, a próxima etapa consiste em codificar o algoritmo para que ele possa criar as pastas que posteriormente irão armazenar as coordenadas dos dados obtidos, para isso foi feito uma automação responsável por criar as pastas com os nomes dos sinais, e dentro de cada pasta criou-se outras pastas incumbidas ordenar os arquivos de cada frame capturado pelas imagens do OpenCV.

Como definido no objetivo deste trabalho, foram mapeados 10 sinais, sendo eles: Bom dia, boa tarde, boa noite, por favor, desculpa, obrigado, ajudar, entender, olá e parabéns. Para fins de evitar redundâncias desnecessárias, optou-se por parametrizar a quantidade de repetições de captura de frame em 30 ciclos, sendo cada ciclo composto por 30 capturas, desta forma obtém-se uma gama de 900 capturas por sinal, como a lista de sinais somam 10 sinais, totalizam-se 9000 frames por rodadas de inserção.

Cada frame é constituído de 21 pontos de articulações para cada mão e cada ponto possui 3 valores cardinais (X, Y e Z), totalizando 126 pontos de referência. Para a construção da base de dados, três pessoas foram convidadas a realizar os sinais que seriam usados para estruturar a base de dados, duas pessoas deficientes auditivas que possuem a Libras como língua materna e uma pessoa ouvinte que possui conhecimentos em Libras, desta forma a base de dados será constituída de 27000 frames.

2.5 Critérios para avaliação da pesquisa

Para validar a credibilidade da base de dados, esta foi submetida a três modelos distintos de aprendizado de máquina, sendo eles o LSTM (Long Short Term Memory), o SVM (Support Vector Machine) e por fim o Random Forest, com o objetivo de criar algoritmos capazes de reconhecer os sinais de Libras apresentados nesta pesquisa, em tempo real.

Cada modelo foi estruturado em uma aplicação que apresenta na tela os sinais que espera receber em forma de lista. Estas aplicações fazem uso da câmera da máquina a qual foi instalada, podendo capturar sinais e interpretá-los em palavras.

O teste dos modelos foi realizado por intermédio de uma pessoa voluntária que possui conhecimentos em Libras e foi instruída a realizar os sinais que consistem na base de dados: Bom dia, boa tarde, boa noite, por favor, desculpa, obrigado, ajudar, entender, olá e parabéns. Os sinais foram realizados de maneira pausada, onde a pessoa estava posicionada a frente da câmera a fim de mantê-la centralizada.

A cada sinal apresentado os modelos indicam qual a palavra mais provável aquele sinal representa. As classificações foram anotadas a fim de constituir um relatório de acuracidade dos modelos treinados, podem assim avaliar a base de dados como mais favorável a treinar futuros modelos do algoritmo que obteve melhor resultado.

REFERÊNCIAS

COUTO, Y. S. **Estruturas de dados persistentes**. 2018. 85p. Dissertação (Mestre em Ciências) - Instituto de Matemática e Estatística – USP (Universidade de São Paulo), São Paulo.

CRUZ, A. R. dos S. **UMA ESTRATÉGIA PARA RECONHECIMENTO DE SINAIS DA LÍNGUA BRASILEIRA DE SINAIS UTILIZANDO APRENDIZADO PROFUNDO**. 2020. 78p. Dissertação (Mestre em Informática) - Instituto de Computação - UFAM (Universidade Federal do Amazonas), Manaus.

EDELWEISS, N.; GALANTE, R. **Estruturas de dados**. 1. ed. Porto Alegre : Bookman, 2009.

FELIPE, T. A.; MONTEIRO, M. S. S. **Libras em Contexto: Curso Básico: Livro do Professor**. 6. ed. Brasília : MEC, 2007.

FONSECA, F. F. **VISAO COMPUTACIONAL APLICADA AO RECONHECIMENTO DE IMAGENS RELACIONADAS À LÍNGUA BRASILEIRA DE SINAIS**. 2020. 105p. Monografia (Engenharia Eletrônica e de Computação) - Escola Politécnica, UFRJ (Universidade Federal do Rio de Janeiro), Rio de Janeiro.

GIL, C. A. **Como Elaborar Projetos de Pesquisa**. 3. ed. São Paulo : Atlas, 1991.

MARENGONI, M.; STRINGHINI, D. **Tutorial: Introdução à Visão Computacional usando OpenCV**. 2009. RITA, vol. XVI, Número 1, São Paulo.

MARTINS, R. A. S. **Sistema de contagem de pessoas na biblioteca da FEUP utilizando visão computacional**. 2021. 96p. Mestrado Integrado (Engenharia Eletrotécnica e de Computadores) - Faculdade de Engenharia, FEUP (Universidade do Porto), Porto, Portugal.

NASCIMENTO, R.; FONTENELE, J.; BALUZ, R.; MOREIRA, R.; TEIXEIRA, S.; TELES, A. **Soluções de Aprendizado de Máquina para Estimar em Tempo Real a Pose Humana em Aplicações de Saúde** In: **LIVRO DE MINICURSOS ERCEMAPI 2020**. Porto Alegre : Secretaria Regional SBC Piauí, 2020. cap. 6, p. 121-145.

NOGUEIRA, A. **SignPic: Sistema Móvel Para Detecção de Língua Gestual Utilizando Machine Learning**. 2020. 55p. (Master of Science) - Faculdade de Ciências e Tecnologia, UFP (Universidade Fernando Pessoa), Porto, Portugal.

PUGA, S.; RISSETT, G. **Lógica de programação e estrutura de dados, com aplicações em Java**. 2. ed. São Paulo : Pearson Prentice Hall, 2009.

STROBEL, K. **HISTÓRIA DA EDUCAÇÃO DE SURDOS**. 2009. 49p. Licenciatura (Letras-LIBRAS na modalidade a distância) - UFSC (Universidade Federal de Santa Catarina), Florianópolis.

ANEXOS

Obs. Caso seja necessário.

Segundo a NBR 14724, de dezembro de 2005, a **diferença entre Anexo e Apêndice** é que o **Anexo** é um texto ou documento não elaborado pelo autor do Trabalho Científico (monografia, tese, etc.) e o **Apêndice** é um texto ou documento elaborado pelo autor do trabalho.

APÊNDICE

Obs. Caso seja necessário.

Segundo a NBR 14724, de dezembro de 2005, a **diferença entre Anexo e Apêndice** é que o **Anexo** é um texto ou documento não elaborado pelo autor do Trabalho Científico (monografia, tese, etc.) e o **Apêndice** é um texto ou documento elaborado pelo autor do trabalho.