UNIVERSIDADE FEDERAL DE OURO PRETO
ESCOLA DE MINAS
COLEGIADO DO CURSO DE ENGENHARIA DE CONTROLE
E AUTOMAÇÃO - CECAU

FERNANDO TORRES BERNARDO

DESENVOLVIMENTO DE UMA INTERFACE HOMEM-MÁQUINA UTILIZANDO VISÃO COMPUTACIONAL PARA ACESSIBILIDADE DIGITAL

MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Ouro Preto, 2019

FERNANDO TORRES BERNARDO

DESENVOLVIMENTO DE UMA INTERFACE HOMEM-MÁQUINA UTILIZANDO VISÃO COMPUTACIONAL PARA ACESSIBILIDADE DIGITAL

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para a obtenção do Grau de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Agnaldo José da Rocha Reis

Ouro Preto
Escola de Minas – UFOP
2019

B523d Bernardo, Fernando Torres.
Desenvolvimento de uma interface homem-máquina utilizando visão computacional para acessibilidade digital [manuscrito] / Fernando Torres Bernardo. - 2019.

53f.:

Orientador: Prof. Dr. Agnaldo José da Rocha Reis.

Monografia (Graduação). Universidade Federal de Ouro Preto. Escola de Minas. Departamento de Engenharia de Controle e Automação e Técnicas Fundamentais.

1. Visão Computacional. 2. Tecnologias Assistivas. 3. Reconhecimento de Gestos. 4. Interface Homem Máquina. 5. Aprendizado de Máquina. I. Reis, Agnaldo José da Rocha. II. Universidade Federal de Ouro Preto. III. Titulo.

CDU: 681.5

Catalogação: ficha.sisbin@ufop.edu.br

A comissão avaliadora constituída pelos professores Agnaldo José da Rocha Reis, Sílvia Grasiella Moreira Almeida e Eduardo José da Silva Luz atesta que a monografia intitulada "Desenvolvimento De Uma Interface Homem-Máquina Utilizando Visão Computacional Para Acessibilidade Digital" foi defendida e aprovada em 18 de julho de 2019.

Prof. Dr. Agnaldo José da Rocha Reis – Orientador

Profa. Dra. Sílvia Grasiella Moreira Almeida – Professora Convidada

Prof. Dr. Eduardo José da Silva Luz – Professor Convidado

AGRADECIMENTOS

Quero agradecer primeiramente a Deus por essa conquista e dedicá-la aos meus pais. Obrigado por todo carinho, educação e suporte necessário para me tornar quem eu sou e chegar onde cheguei.

Agradeço à minha namorada Thaís por estar presente durante toda essa jornada, e por me apoiar e me cobrar sempre que precisei. Com você aprendi o valor da dedicação. Thaís, essa e tantas outras conquistas só se fizeram possíveis porque seu apoio e carinho foram a minha maior motivação.

Ao meu irmão Raul, por ser meu exemplo de trajetória, pelos ensinamentos, por me ajudar a trilhar o meu próprio caminho e me apoiar em cada decisão.

Aos meus familiares, minha cunhada Larissa, meus tios e tias, primos e primas e meus avós, por estarem presentes e sempre torcerem pelo meu melhor.

Ao meu orientador professor Doutor Agnaldo José da Rocha Reis, pela orientação e dedicação em me ajudar neste e em outros trabalhos.

Aos alunos do curso de Engenharia de Controle e Automação, em especial aos amigos do período 14.2, pelos momentos e aprendizados que vivemos juntos.

À todos os professores, em especial aos professores do curso de Engenharia de Controle e Automação e aos professores Doutor Eduardo José da Silva Luz e Doutor Gladston Juliano Prates Moreira pelo imenso aprendizado junto ao CSILab.

À república Kasa Cheia, por seu meu lar nesta etapa, por ser minha válvula de escape para as dificuldades encontradas e por ser onde construí amizades que levarei para vida.

Por fim, agradeço a Universidade Federal de Ouro Preto, em especial à Escola de Minas, por todo aprendizado profissional e pessoal adquirido.

“Se eu vi mais longe, foi por estar sobre ombros de gigantes.” (Isaac Newton)

RESUMO

Embora as tecnologias de uso cotidiano tenham avançando enormemente com o passar do tempo, a inclusão digital de pessoas com limitações motoras ainda é uma tarefa difícil. Certos graus de limitações impedem o uso de aparelhos essenciais para o acesso à essas tecnologias, como por exemplo, o mouse, o teclado, celulares e tablets. Por outro lado, as tecnologias assistivas proporcionam diferentes meios de acesso à tecnologia, gerando uma aproximação entre pessoas com limitações especiais e o mundo digital. Neste trabalho foram condensadas técnicas de Visão Computacional e Aprendizado de Máquina de forma a criar uma Interface Homem-Máquina (IHM) livre do uso das mãos. A interface desenvolvida foi adaptada de forma a proporcionar o controle do feed de notícias do Facebook. Utilizando apenas movimentos realizados com a cabeça, o usuário é capaz de navegar, expandir e curtir as publicações.

Palavras-chaves: Tecnologias Assistivas, Visão Computacional, Reconhecimento de Gestos, Interface Homem Máquina, Aprendizado de Máquina.

ABSTRACT

Although everyday tecnhologies has been making big progress trought time, digital inclusion is still a difficult task. Certain degrees of motor limitations prevent the use of essential devices for technology acessing, such as mouse, keyboards, celphones and tablets. On the other hand, assistive technologies provide different means of consuming certain technologies, reducing the gap between people with motor disabilities and the digital world. In this work, several Computer Vision tecniques were condensed in order to create an Human Machine Interface free of the use of the hands. The developed interface is able to control the Facebook news feed. Using only head moviments the user is able to browse, expand and like the posts.

Key-words: Assistive Technology, Computer Vision, Gesture recognition, Human Machine Interface, Machine Learning.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CNN	Convolutional Neural Network
DL	Deep Learning
FPS	Frames por Segundo
GPU	Graphics Processing Unit
IA	Inteligência Artificial
IBGE	Instituto Brasileiro de Geografia e Estatística
IHM	Interface Homem-Máquina
ILSVRC	ImageNet Large-Scale Visual Recognition Challenge
ML	Machine Learning
MLP	Multilayer Perceptron
PDI	Processamento Digital de Imagens
PFF	Pontos Fiduciais Faciais
PnP	Perspective-n-Point
ReLU	Rectified Linear Units
SVM	Support Vector Machine
TA	Tecnologias Assistivas
VC	Visão Computacional

LISTA DE SÍMBOLOS

u	Posição angular de *Roll*
v	Posição angular de *Pitch*
w	Posição angular de *Yaw*
$\dot{u}$	Velocidade angular de *Roll*
$\dot{v}$	Velocidade angular de *Pitch*
$\dot{w}$	Velocidade angular de *Yaw*
Y_{max}	Limiar máximo para o angulo de rotação v
Y_{min}	Limiar mínimo para o angulo de rotação v
X_{max}	Limiar máximo para o angulo de rotação w
X_{min}	Limiar máximo para o angulo de rotação w

SUMÁRIO

1 INTRODUÇÃO

A tarefa de identificar e interpretar movimentos é um hábito natural para o ser humano. Porém, transferir essa habilidade para um computador não é uma tarefa trivial. Para tanto, é necessário que o computador perceba elementos visuais no ambiente e de alguma forma interprete suas trajetórias ao longo do tempo. A Visão Computacional (VC) é o ramo da inteligência artificial responsável por fornecer a habilidade de interpretar o conteúdo de uma imagem ao computador. Esta abordagem pode ser descrita como uma transformação de dados de uma imagem estática ou de vídeo em uma decisão ou em uma nova representação (BRADSKI; KAEHLER, 2008). Foi realizado um levantamento com relação ao número de trabalhos envolvendo o tema VC através da biblioteca digital IEEE Xplore. Os dados foram levantados através da opção de busca avançada e utilizando o descritor em inglês *Computer Vision* como filtro. O intervalo de tempo analisado englobou os anos de 1975 até 2018.

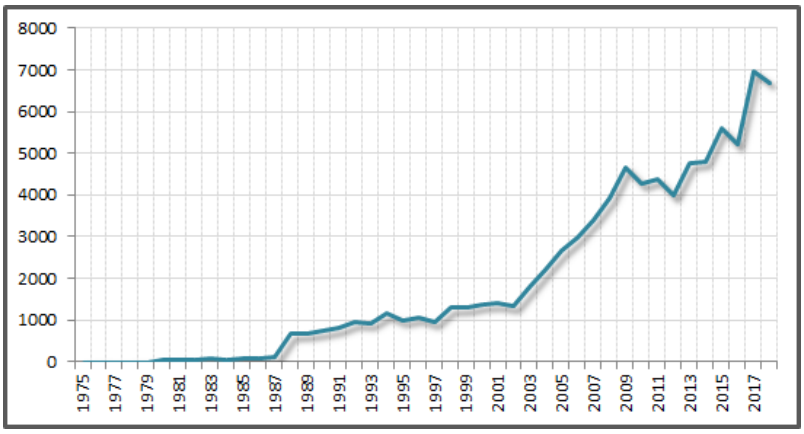

Figura 1 – Evolução do número de pesquisas envolvendo o tema *Computer Vision* dos anos 1975 até 2018.

Como mostrado no gráfico exposto na figura 1, o número de pesquisas envolvendo visão computacional cresceu na década de 80 e mais expressivamente nos anos 2000. O aumento do interesse pelo ramo é reflexo de alguns trabalhos relevantes para o tema publicados nesse período. O desenvolvimento da subárea da IA, denominada Aprendizado de Máquina (ML do inglês *Machine Learning*) , possibilitou o desenvolvimento de modelos de VC capazes de superar os melhores resultados alcançados até o momento em desafios de reconhecimento de padrões. O uso de algoritmos de retropropagação - em inglês *backpropagation* - como apresentado em (LECUN et al., 1989), possibilitou o ajuste automático de parâmetros utilizando exemplos de entradas e suas respectivas respostas esperadas. Essa técnica junto com a evolução das CPU e GPU, possibilitou o desenvolvimento de modelos mais complexos permitindo elevar o número de camadas e, consequentemente, o número de parâmetros. Com o aumento de abordagens

utilizando modelos denominados "profundos"pelo volume de parâmetros, criou-se uma sub área do ML chamada Aprendizagem Profunda (DL do inglês *Deep Learning*).

A principal abordagem que impulsionou os trabalhos de DL foi o uso de Redes Neurais Convolucionais (CNN do inglês *Convolutional Neural Networks*) propostas em (FUKUSHIMA, 1980). O uso de CNN conseguiu superar os recordes de desempenho em desafios de alta dificuldade como ILSVRC (ImageNet Large-Scale Visual Recognition Challenge) (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Essas redes assemelham-se as redes neurais artificiais clássicas como o Perceptron Multicamada (MLP do inglês Multilayer Perceptron), porém, a operação de convolução substituiu a multiplicação matricial padrão, reduzindo o número de parâmetros ajustáveis. Além disso, as imagens podem ser carregadas diretamente para a rede evitando o pré-processamento para extração de características (LIU et al., 2017). Devido a alta capacidade de identificar padrões não lineares apresentada por uma CNN, tonaram-se diversas suas aplicações. São exemplos de aplicação: detecção de objetos, classificação de imagens, classificação de sinais, reconhecimento de gestos, reconhecimento facial e previsão de séries temporais. Porém, para ajustar os milhares de parâmetros contidos em uma CNN de forma automática é necessário uma grande quantidade de exemplos categorizados. Por isso, um fator essencial que explica o número de trabalhos utilizando essas redes foi a construção de grandes bases de dados categorizadas como, por exemplo, a *Imagenet* (DENG et al., 2009).

Embora os resultados utilizando técnicas de DL alcancem recordes de desempenho, essa técnica ainda demanda um alto custo computacional. Os computadores específicos para tarefas de DL em larga escala, utilizam placas de processamento gráfico (GPU), exclusivas para acelerar o processamento. Porém, muitos dos computadores pessoais não possuem infraestrutura suficiente para o uso dessa abordagem em tempo real. Por isso, muitas vezes técnicas com baixo nível de processamento se adequam melhor a certos problemas. A detecção facial em aparelhos com baixa capacidade de processamento em tempo real, por exemplo, é uma tarefa que emprega classificadores em cascata, como proposto pelos autores Paul Viola e Michael Jones em (VIOLA; JONES, 2004). A técnica também conhecida como ViolaJones, utiliza máscaras para extração de características baseadas em filtros de *Haar* e uma estrutura em cascata para reduzir o tempo de processamento necessário.

Sistemas capazes de realizar tarefas de reconhecimento de imagens possuem uma ampla gama de aplicação como, por exemplo, o desenvolvimento de carros autônomos, controle de acesso via biometria, monitoramento de locais públicos, monitoramento de exposição de logomarcas e controle de equipamentos via gestos. Portanto, é necessário compreender as especificações de cada aplicação para escolher o equilíbrio entre o desempenho da técnica e seu custo computacional. Este trabalho visa combinar técnicas de VC e ML de forma a reconhecer gestos visuais realizados com a cabeça de um usuário com limitações motoras, em tempo real, como uma alternativa de interação com o computador. Desta forma, para simular o desempenho do sistema, foi realizada uma avaliação utilizando a base de dados BUHMAP.

1.1 Justificativas e Relevância

Uma vez dada a capacidade de interpretar gestos humanos aos computadores através de imagens, podemos considerar o ramo da visão computacional como alternativa para tornar mais natural a interação humana com as tecnologias utilizadas no cotidiano. Facilitar a interação digital impactaria diretamente a sociedade atual, uma vez que, a evolução digital está avançando bruscamente e com isso, modificando diretamente a forma como vivemos. Porém, inclusão digital de pessoas com limitações motoras ainda é um problema a ser solucionado. Segundo a Pesquisa Nacional da Saúde do IBGE de 2013, 1,3% da população do Brasil declarou possuir deficiência física e desta parcela, 46,8% possui grau intenso ou muito intenso de limitação (ESTATÍSTICA, 2015). Muitas vezes essas limitações impossibilitam o uso de instrumentos manuais como o mouse, teclado e o celular, fazendo com que essa parcela da população seja impossibilitada de usufruir amplamente das tecnologias atuais.

As Tecnologias Assistivas (TA) surgiram como as tecnologias desenvolvidas especialmente para dar mais autonomia e qualidade de vida. Os trabalhos envolvendo TA muitas vezes alcançam soluções para os problemas de mobilidade, porém, com um custo computacional elevado atrelado.

Desse modo, concentra-se neste trabalho, esforços na tarefa de explorar técnicas de VC como alternativa para maior inclusão de pessoas com limitações motoras no mundo digital, construindo uma interface com usuário utilizando somente processamento de imagem de forma viável para computadores de uso pessoal. Os gestos a serem reconhecidos serão realizados com a cabeça, tornando a interface independente da mobilidade dos membros superiores do usuário.

1.2 Metodologia

Primeiramente foi realizado um estudo exploratório acerca do tema VC para compreensão de diversas técnicas e construção de uma base teórica para o trabalho. Após o levantamento do referencial teórico, foram selecionadas técnicas de VC e ML, além disso, foi proposta uma abordagem para o reconhecimento de gestos. Com o intuito de avaliar a abordagem proposta, foi desenvolvido um protótipo de interface com o usuário para controle de uma tarefa de uso cotidiano. O sistema proposto neste trabalho foi desenvolvido utilizando a linguagem de programação Python 3.6, em um notebook de uso pessoal. O protótipo desenvolvido é composto por dois módulos principais, Visão Computacional (Módulo 1) e Análise de Sinais (Módulo 2). O fluxograma de funcionamento do sistema está detalhado no capítulo 3. Além disso, os comandos gerados pelo protótipo, foram adaptados para controlar o feed de notícias do Facebook. Por fim, foram realizados experimentos práticos de forma a avaliar o desempenho da abordagem proposta, nesta etapa foi utilizada uma base de dados retirada da literatura.

1.3 Objetivos

O objetivo principal deste trabalho é desenvolver uma interface homem-máquina, controlada apenas por movimentos realizados com a cabeça. O intuito desta interface é proporcionar um ambiente de controle independente do uso das mãos ou de aparelhos adicionais, sendo assim, uma alternativa mais acessível para pessoas com limitações motoras dos membros superiores.

Como objetivos específicos estão:

- Identificar a posição da face do usuário, para cada frame de um vídeo, em tempo real.
- Estimar a orientação da cabeça do usuário e monitorar sua trajetória ao longo do tempo.
- Classificar e traduzir padrões presentes em séries temporais para sinais de comando.
- Desenvolver uma interface de controle simples e independente de dispositivos externos ao computador.
- Comunicar a interface a um aplicativo de uso cotidiano a partir dos movimentos detectados.

1.4 Organização e estrutura

O restante deste trabalho está organizado da seguinte forma. No capítulo 2, está apresentada a revisão literária de trabalhos relacionados com o tema da pesquisa, utilizada como referencial teórico para confecção do trabalho. No capítulo 3, detalha-se a abordagem e as técnicas utilizadas para o desenvolvimento do sistema proposto. No capítulo 4, são detalhados os experimentos realizados para validar o desempenho das técnicas utilizadas, além disso, também estão detalhados os resultados encontrados nos experimentos realizados durante este trabalho. Na Conclusão resume-se as conclusões encontradas neste trabalho e fala-se as propostas para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Reconhecimento de Gestos

Em (MITRA; ACHARYA, 2007) os autores descrevem os gestos como movimentos corporais expressivos e significativos que envolvem movimentos físicos dos dedos, mãos, braços, cabeça, face ou corpo com a intenção de transmitir informações significativas ou interagir com o ambiente. Um gesto também pode ser percebido pelo ambiente como uma técnica de compressão para a informação a ser transmitida e subsequentemente reconstruída pelo receptor. Os gestos são divididos entre estáticos e dinâmicos. Os estáticos são aqueles que podem ser percebidos a partir de imagens isoladas (o usuário assume uma posição estática). Enquanto os dinâmicos são aqueles que dependem da associação das informações de uma sequência de imagens (o usuário executa uma série de movimentos).

Reconhecer estes gestos é uma tarefa complexa que envolve muitos aspectos, como análise e modelagem de movimento, reconhecimento de padrões, aprendizado de máquina e estudos psicolinguísticos (WU; HUANG, 1999). O funcionamento do sistema deve ser capaz de identificar e monitorar um objeto de interesse, extrair características representativas dos dados monitorados e por fim classifica-las em categorias pré-definidas. Alguns trabalhos utilizam aparatos sensoriais para capturar os dados necessários, no entanto, muitas vezes esses dispositivos elevam o custo final do sistema. Por isso, o uso de técnicas de VC e processamento digital de imagens (PDI) surgem como alternativa para desenvolvimento de sistemas mais acessíveis e menos invasivos.

As aplicações de sistemas que utilizam algum nível de reconhecimento de gestos são diversas. Em (QUEK, 1996) os autores apresentaram um aplicativo chamado *FingerMouse* para reconhecer os movimentos dos dedos em 2 dimensões, utilizando os gestos identificados como entrada para comandos na área de trabalho. Em (CROWLEY et al., 1995) os autores também desenvolvem uma aplicação chamada *FingerPaint* para rastrear a trajetória do dedo do usuário. A ferramenta foi utiliza como um dispositivo de realidade aumentada para desenhar utilizando o movimento do dedo. Em (ZELLER et al., 1997) os autores apresentaram um ambiente virtual para modelagem biomolecular de grande escala. Este sistema permite a modelagem interativa de biopolímeros, utilizando programas de simulação de gráficos moleculares 3D, e reconhecendo gestos com a mão para controle dos gráficos. Em (NISHIKAWA et al., 2003), os autores desenvolveram uma interface denominada *FAce MOUSe* para controle de posição de um laparoscópio em cirurgias. Os autores afirmam que o sistema permite controlar o laparoscópio de forma precisa utilizando somente gestos faciais, evitando o uso de acessórios extras.

2.2 Detecção Facial

A detecção facial é um ramo amplamente explorado na comunidade acadêmica desde a década de 90 e tem como objetivo encontrar em uma imagem digital as regiões correspondentes ao rosto de um indivíduo. Em (HJELMÅS; LOW, 2001) os autores definem a tarefa de detecção facial através do seguinte enunciado: Dada uma imagem estática ou um vídeo, detecte e localize um número desconhecido (se houver) de faces. Em (HSU; ABDEL-MOTTALEB; JAIN, 2002) os autores propuseram um algoritmo capaz de detectar a região facial utilizando detecção de pele. O tom presente na pele humana encontra-se em regiões específicas da distribuição RGB quando não estão presentes variações de iluminação, podendo ser segmentado em uma imagem digital. Uma vez detectada uma região na imagem contendo pele, os autores compararam formatos relacionados aos olhos e a boca para confirmar que naquela região contém uma face. Os autores utilizaram técnicas de compensação de luminosidade e a transformação não linear de RGB para YCbCr, tornando o método mais robusto a varição de luminosidade.

Em 2003, Paul Viola e Michael J. Jones propuseram em (VIOLA; JONES, 2004), o algoritmo conhecido como Viola Jones para detecção facial em tempo real. Os autores desenvolveram o método utilizando extratores de características baseados nas características de *Haar* e propondo três novos conceitos: imagem integral, um algoritmo de aprendizado de classificadores baseado no algoritmo *AdaBoost* e a estrutura em cascata. O método é capaz de detectar múltiplas faces na imagem com alta taxa de acerto mantendo o tempo de processamento baixo. Os autores afirmam em (ZHANG; ZHANG, 2010) que o algoritmo pode ser considerado de maior impacto para detecção facial nos anos 2000 e foi responsável por tornar viável o amplo uso de detecção facial no mundo real. A técnica possibilitou aplicações como identificação de faces em câmeras digitais e algoritmos de agrupamento em softwares de organização de fotos.

Em (JIANG; LEARNED-MILLER, 2017) os autores treinaram uma rede convolucional baseada em regiões (R-CNN) para a tarefa de detecção facial, superando os resultados das 11 melhores técnicas propostas até 2015 para duas bases de dados. Apesar dos resultados mostrarem a superioridade em desempenho dessas redes, o custo computacional para realizar uma inferência ainda é muito alto. Utilizar essas redes para detecção em tempo real requer uma infraestrutura computacional específica, muitas vezes sendo necessário o uso de GPUs (Graphics Processing Unit, ou Unidade de Processamento Gráfico) dedicadas para acelerar o processamento.

2.3 Orientação Facial

A posição da cabeça no espaço pode ser representada através da orientação de três ângulos principais (*pitch*, *yaw* e *roll*) representados na figura 2. Os movimentos realizados com a cabeça possuem três graus de liberdade, desta forma, estes ângulos são suficientemente representativos para identificação de gestos realizados com a cabeça.

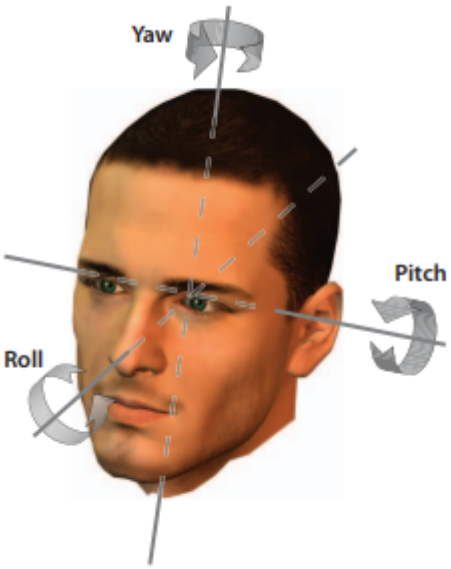

Figura 2 – Representação dos três ângulos utilizados para indicar a orientação da cabeça no espaço, retirado de (MURPHY-CHUTORIAN; TRIVEDI, 2008).

Existem diversas abordagens para a estimativa de pose facial, uma delas parte da comparação da imagem com templates (imagens de exemplo) de posições predeterminadas como proposto em (NIYOGI; FREEMAN, 1996). Porém, este método possibilita uma variedade pequena de poses, uma vez que o custo computacional aumenta significativamente com o aumento do número de templates. Outra abordagem é a construção de detectores faciais específicos para cada pose determinada, em (HUANG; SHAO; WECHSLER, 1998) os autores utilizaram um classificador SVM. Porém, desenvolver diversos classificadores exige uma amostra balanceada e suficientemente genérica para cada uma das poses. Desta forma, a estratégia que parece mais ajustada ao problema é a regressão da imagem de entrada nos valores reais de orientação. Em (OSADCHY; CUN; MILLER, 2007) os autores utilizaram uma CNN como modelo de regressão.

Uma alternativa para estimativa da orientação da cabeça de forma simples e rápida é a associação de informações de um modelo tridimensional genérico. Considerando pontos de referência da face na imagem como projeções de um modelo genérico tridimensional, pode-se estimar a posição angular da face através da solução de um problema PnP (*Perspective-n-Point*), utilizado para determinar a perspectiva da câmera em relação a um objeto, como apresentado em (OHAYON; RIVLIN, 2006). Os pontos de referência a serem utilizados devem possuir características específicas da face, por isso são escolhidos pontos especiais denominados pontos fiduciais. Um ponto fiducial facial (PFF) é definido por uma posição específica na face humana que esteja presente na maioria das observações da mesma. São exemplos desses pontos, os contornos e centros das sobrancelhas, dos olhos, nariz e boca (KATSIKITIS, 2003).

Para seleção dos pontos de referência os autores utilizaram em (AKAKIN; SANKUR, 2011) um classificador SVM (*Support Vector Machine*). As regiões candidatas à classificação

são pré-processadas através da Transformada Discreta de Cosseno. Os autores realizaram ainda o refinamento das posições estimadas através de operações estatísticas. Porém, os autores não indicaram o tempo de processamento necessário. Em (KAZEMI; SULLIVAN, 2014) os autores propuseram um método capaz de identificar as coordenadas x e y de 194 pontos da face em milissegundos. Os pontos estão dispostos no contorno dos olhos, sobrancelhas, boca, nariz e queixo. O método utiliza modelos de regressão associados em cascata para obter as coordenadas.

2.4 Classificação de sinais

A tarefa do componente **classificador** de um sistema é usar o vetor de características fornecido pelo extrator de características para atribuir ao objeto uma categoria (DUDA; HART; STORK, 2012). Em geral, os classificadores são desenvolvidos a partir de uma amostra de exemplos para o problema em questão e os seus respectivos rótulos. Uma parcela dos exemplos são utilizados para ajustar os parâmetros da técnica escolhida, este procedimento é denominado etapa de treinamento. Outra parcela de exemplos é então utilizada para avaliar o desempenho do classificador gerado, este procedimento é denominado etapa de testes.

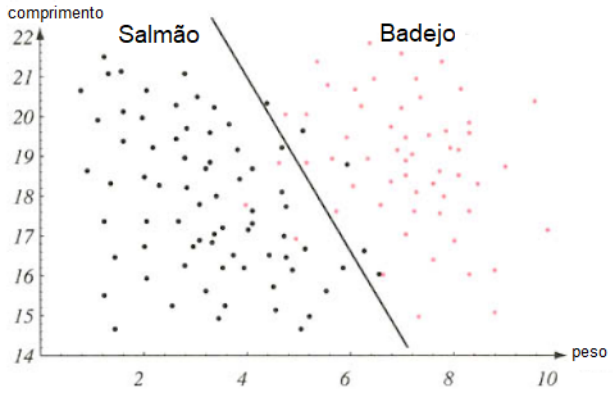

Figura 3 – Exemplo de classificação dos peixes salmão e badejo utilizando o peso e comprimento, adaptado de (DUDA; HART; STORK, 2012).

Um exemplo para o problema de classificação é representado na figura 3, retirada do livro (DUDA; HART; STORK, 2012). O problema consiste em definir a espécie de um peixe a partir do peso e do seu comprimento. O exemplo mostra que em alguns casos as características utilizadas estão muito próximas dificultando a classificação. O método de classificação a partir de uma reta delimitadora, por exemplo, não seria capaz de classificar corretamente todos os peixes.

Para o reconhecimento de gestos dinâmicos realizados com a cabeça, é necessário monitorar as características identificadas nas etapas de detecção facial e estimativa de orientação da cabeça ao longo do tempo, gerando representações temporais. Algumas abordagens utilizam vetores de características contendo parâmetros observados para representar o movimento. Como em (ARI; UYAR; AKARUN, 2008), onde os autores concatenaram as amplitudes máxima e

mínima da diferença entre as coordenadas x e y dos pontos de referência, para cada frame durante o vídeo, em relação as suas respectivas posições iniciais. Porém, nesta abordagem o vetor de características se torna dependente da posição inicial da face no vídeo, além de, serem perdidas informações de trajeto. Por isso, algumas abordagens representam a trajetória das características monitoradas ao longo do vídeo em sinais unidimensionais. Desta forma, a tarefa de classificação passa a ser a um problema de reconhecimento de padrões em sinais, ou séries temporais.

Em (ERDEM; SCLAROFF, 2002) os autores representaram a trajetória da cabeça ao longo do tempo concatenando as posições dos pontos de referência para cada instante do vídeo em múltiplas séries temporais. A partir da análise dos sinais gerados, os autores identificaram padrões que que representam os movimentos de *Sim* e *Não* com a cabeça, como representado na figura 4.

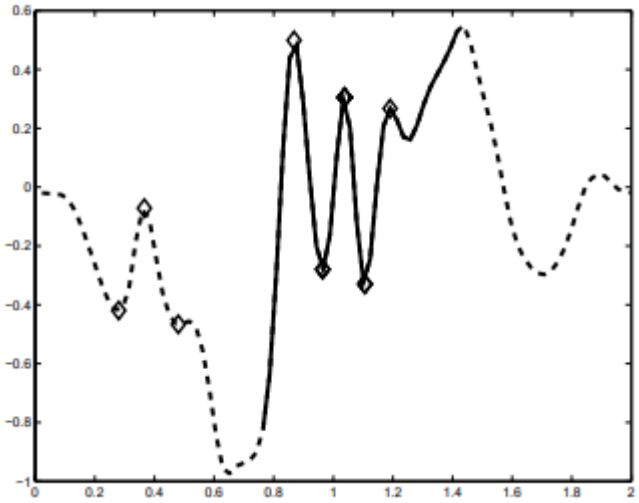

Figura 4 – Representação do padrão de movimento identificado para o gesto *Sim*, retirado de (ERDEM; SCLAROFF, 2002)

A representação em sinais se mostra uma boa abordagem por acumular uma quantidade muito grande de informação. Porém, toda essa informação tem um custo para ser processada e, muitas vezes, apresentam ruídos. Por isso, para evitar o processamento desnecessário de dados, em (ALTHOFF et al., 2005) os autores propuseram uma etapa de detecção antes da classificação dos gestos. No trabalho eles utilizaram a velocidade de trajetória dos pontos de referência para identificar um possível movimento significativo. A velocidade é calculada a partir da derivada do sinal formado pelas posições dos pontos de referência da face ao longo do tempo. O início e fim da faixa a ser considerada são definidos a partir de uma série de regras, como magnitude mínima de velocidade e tempo mínimo de execução. Os autores afirmam alcançar uma taxa de 98% de acerto para detecção dos gestos.

2.5 Redes Neurais Convolucionais

Recentemente as redes neurais convolucionais alcançaram resultados impressionantes em tarefas como detecção, verificação e classificação. Uma das principais razões para o desempenho das CNNs é sua capacidade de aprender representações complexas usando suas camadas convolucionais (CUI; CHEN; CHEN, 2016). Essas camadas substituem as camadas completamente conectadas das redes neurais clássicas, o que reduz consideravelmente a complexidade computacional da rede, permitindo elevar o número de camadas. As redes convolucionais combinam três ideias arquitetônicas para garantir algum grau de invariância, de deslocamento e distorção: campos receptivos locais, pesos compartilhados (ou replicação de peso) e, às vezes, subamostragem espacial ou temporal (LECUN; BENGIO et al., 1995).

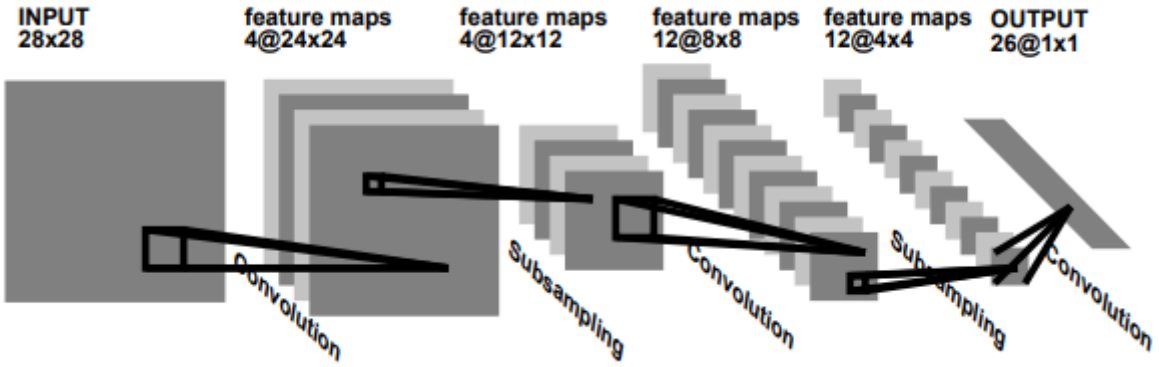

Figura 5 – Representação de uma CNN para reconhecimento de caligrafia, retirado de (LECUN; BENGIO et al., 1995)

A CNN representada na figura 5 foi treinada para reconhecer os caracteres de dígitos numéricos feitos a mão. Para isso, a rede recebe como entrada uma imagem de 28x28 pixeis e retorna um vetor de 26 posições. Segundo os autores, apesar da rede possuir 100.000 conexões, existem apenas 2.600 parâmetros a serem ajustados. As camadas 1,3 e 5 são camadas convolucionais. Na figura 6 está ilustrado a operação de convolução de uma matriz pela máscara $k = [0, 1, 2; 2, 2, 0; 0, 1, 2]$ de tamanho 3x3. Enquanto as camadas 2 e 4 utilizam a operação de subamostragem *maxpooling* considerando uma região de tamanho 2x2 e um passo de tamanho 2. Na figura 7 está ilustrado o resultado da operação de *maxpooling* usando uma região de tamanho 3x3 e um passo de tamanho 1.

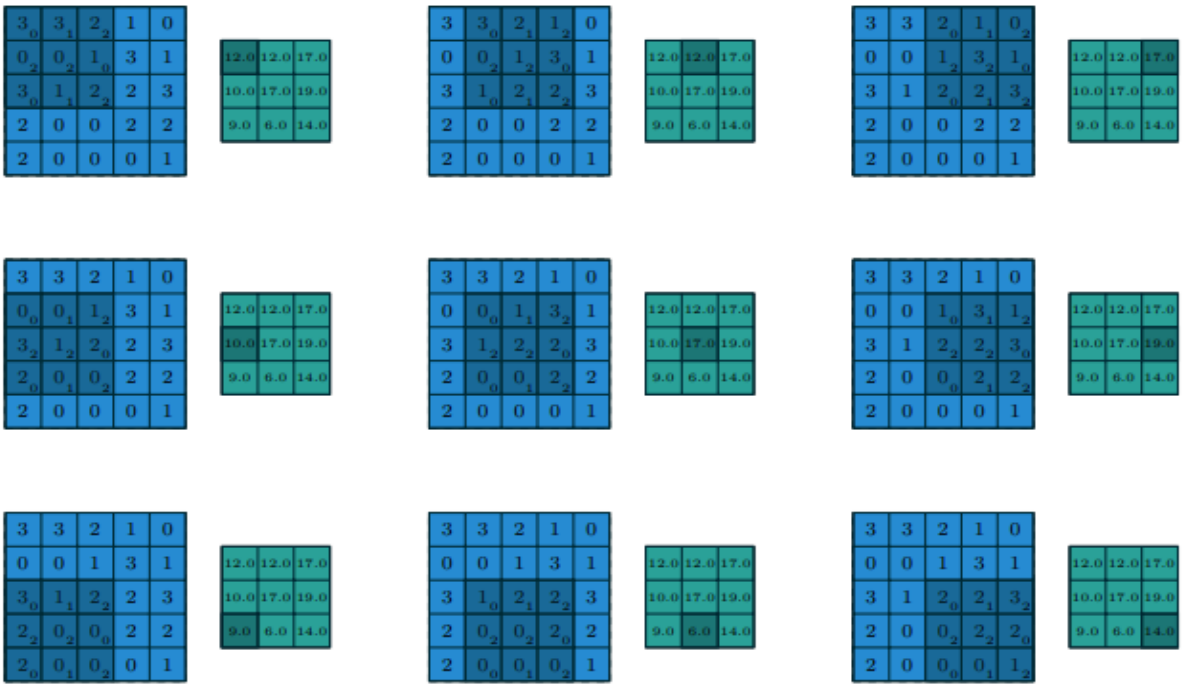

Figura 6 – Exemplo para operação de convolução discreta, retirado de (DUMOULIN; VISIN, 2016).

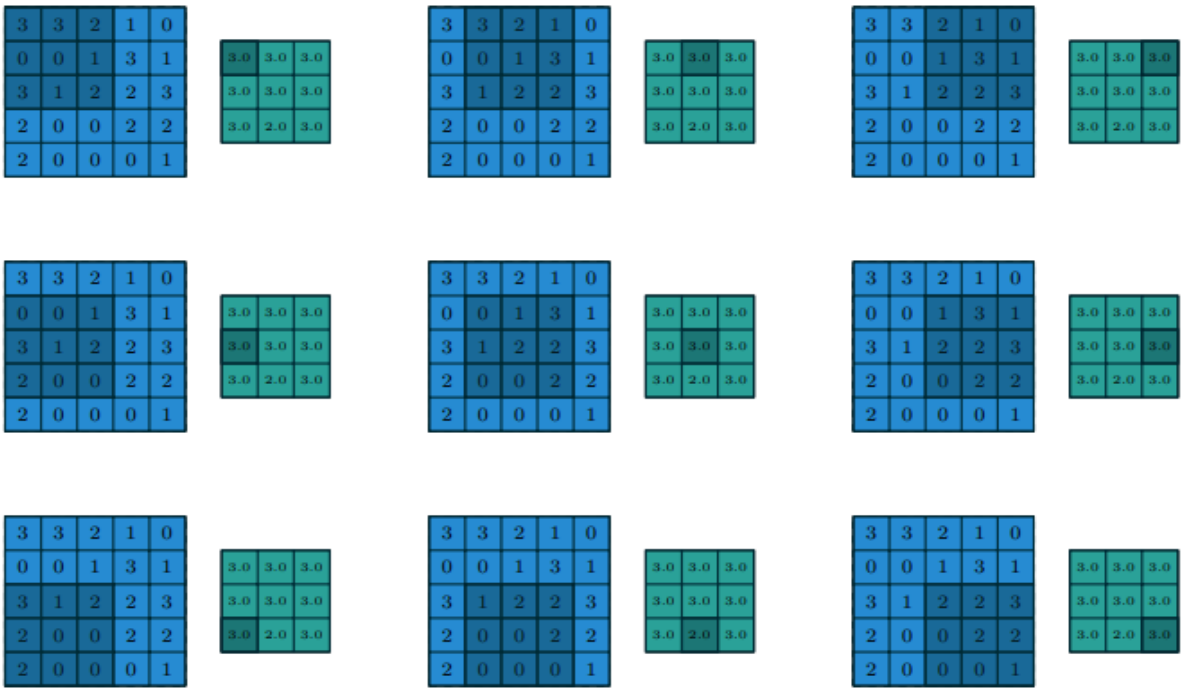

Figura 7 – Exemplo para operação de *maxpooling* com uma região de 3x3 e um passo de tamanho 1, retirado de (DUMOULIN; VISIN, 2016).

Em (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) os autores adicionaram camadas completamente conectadas, equivalentes as camadas de um MLP, ao final da CNN e aumentaram as camadas intermediárias. O resultado alcançou primeiro lugar com uma boa margem no desafio ILSVRC, proposto em 2012, através da base de dados *ImageNet* (DENG et al., 2009). Esse resultado expôs para a comunidade acadêmica a capacidade de uma CNN e elevou a um novo patamar o desempenho de computadores em tarefas como a de classificação. Nos anos que seguiram estes trabalhos, o número de aplicações e diferentes propostas para o uso de *deep learning* cresceu de forma exponencial. Atualmente as CNNs apresentam resultados surpreendentes em tarefas como detecção de objetos (LIU et al., 2016), verificação facial (TAIGMAN et al., 2014), classificação de audio (LEE et al., 2009) e classificação de séries temporais (ZHENG et al., 2014).

2.6 Tecnologias Assistivas

As tecnologias assistivas (TA) são definidas em (COOK; POLGAR, 2014) como tecnologias tradicionais inclusivas e aquelas desenvolvidas especificamente para pessoas com alguma limitação. Os autores afirmam também que a área de estudo envolvendo TA é uma das muitas oportunidades necessárias para reduzir a influência incapacitante de muitos ambientes. São diversas as aplicações dessas tecnologias, como: Auxílio à locomoção, assistência a comunicação, interação com aparelhos digitais, cuidados domésticos e auxílio em exercícios físicos.

Muitos dos trabalhos envolvendo TA fazem uso de acessórios, que são capazes de fornecer dados à um software ou são utilizados para algum tipo de assistência ao usuário. Um exemplo é o sistema desenvolvido em (ZHU et al., 2014), onde diversos sensores são utilizados como acessórios e auxiliam pessoas com limitações visuais na locomoção. Outro sistema proposto em (BÄCHLIN et al., 2010), também utiliza a ideia de vestir sensores para ajudar pacientes diagnosticados com Parkinson a caminhar. Porém, esses acessórios utilizados muitas vezes possuem valor agregado alto, aumentando o custo final do sistema.

Propostas utilizando técnicas de VC surgem como alternativa para eliminar a necessidade de aparelhos e acessórios além do computador, aumentando a viabilidade de distribuição dessas tecnologias. Em (JIA et al., 2007), os autores projetaram um sistema capaz de controlar uma cadeira de rodas elétrica a partir de gestos com a cabeça. O sistema utilizava um classificador em cascata para detectar a face e a técnica de *Template Matching* na região do nariz para reconhecer a posição do rosto. O usuário conseguia acelerar, frear e virar movimentando a cabeça para a direção relativa a cada ação. Em (BETKE; GIPS; FLEMING, 2002), os autores desenvolveram uma interface de controle para o mouse, utilizando movimentos com a cabeça. O sistema monitora uma região do rosto, escolhida pelo usuário, ao longo do tempo. Desta forma, o movimento do mouse acompanhará o movimento monitorado pela webcam.

Além de proporcionar diferentes formas de interagir com o computador, o reconhecimento de gestos visuais também são utilizados para comunicação entre usuários. Um exemplo

dessa aplicação é o aplicativo de tradução automática da língua de sinais da India para língua inglesa desenvolvido em (MADHURI; ANITHA; ANBURAJAN, 2013). Outro exemplo é o classificador de gestos faciais para a língua de sinais da Turquia apresentado em (ARI; UYAR; AKARUN, 2008).

3 DESENVOLVIMENTO

A organização para o protótipo desenvolvido foi dividida em dois módulos principais, como representado na figura 8. Os comandos poderão ser gerados a partir de gestos estáticos ou dinâmicos. Desta forma, um comando será executado a partir da identificação de uma posição específica esperada na saída do módulo 1, ou a partir de uma classificação esperada na saída do módulo 2. Cada um dos módulos, assim como as técnicas utilizadas em cada uma das suas etapas, estão detalhados neste capítulo.

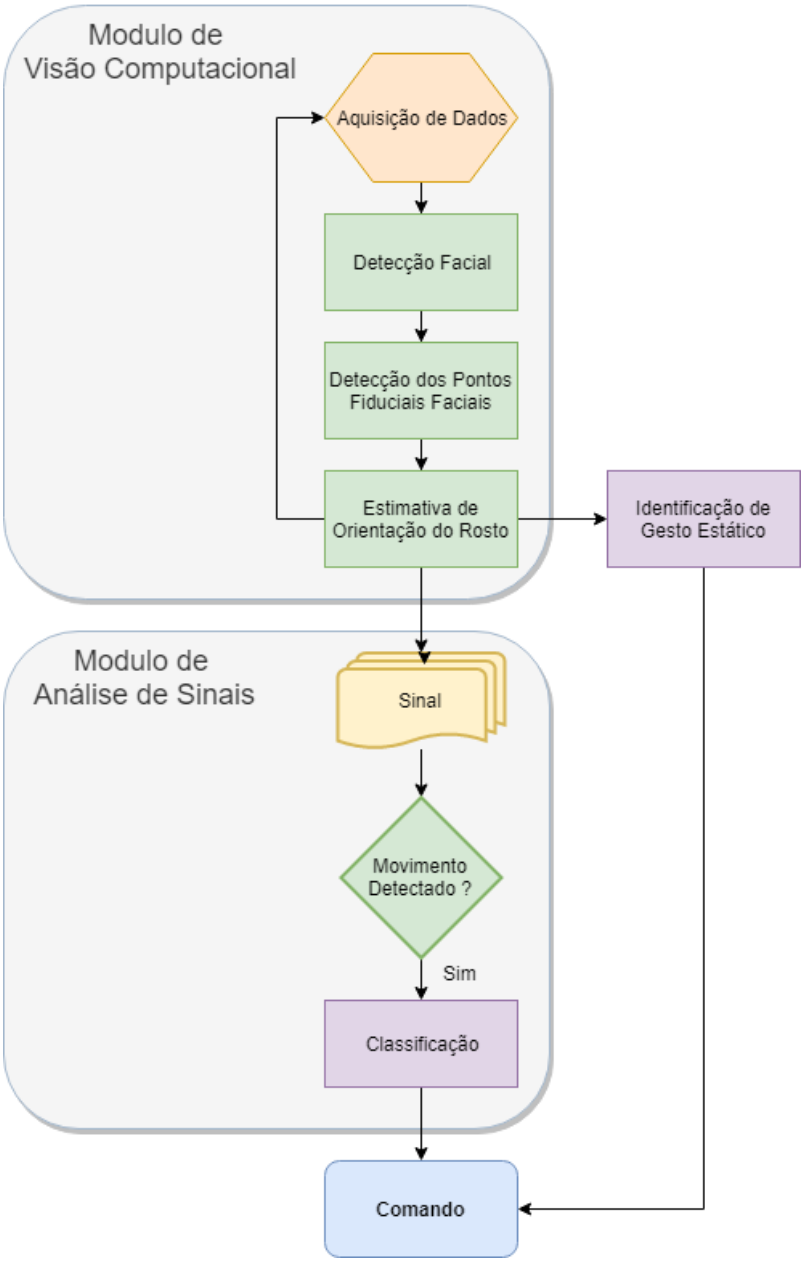

Figura 8 – Fluxo de funcionamento proposto.

3.1 Módulo de Visão Computacional

Este módulo é responsável por estimar a orientação da cabeça do usuário ao longo do vídeo, partindo da imagem digital. Para isso o módulo conta com 4 etapas: Aquisição de dados, Detecção facial, Detecção dos PFF e Estimativa de Orientação. Desta forma, o modulo iniciará o processamento capturando a imagem digital e retornará três ângulos que descrevem a orientação da cabeça do usuário. As etapas funcionam em um loop sequencial, de forma que ao fim da etapa de estimativa de orientação, uma nova aquisição será feita, dando inicio ao processamento de um novo frame.

3.1.1 Aquisição de Dados

A aquisição de dados é a primeira etapa do loop de execução do módulo e consiste em capturar as imagens digitais por meio de uma câmera conectada ao computador. A comunicação do sistema com a câmera é feita por meio da biblioteca *OpenCV*. A imagem adquirida é representada por meio de uma matriz com dimensão 480x640 pixeis, onde cada pixel possui 3 canais no formato RGB. A taxa de aquisição alcançada pelo sistema foi de 12 FPS.

3.1.2 Detecção facial

A tarefa de detectar faces na imagem é realizada através de um modelo pré-treinado disponibilizado pela bilioteca *OpenCV*. O modelo é baseado na abordagem apresentada em (VIOLA; JONES, 2004) e utiliza uma série de classificadores baseados em características de *Haar* (Figura 11). Como a técnica leva em consideração apenas a intensidade dos pixeis para a detecção, é necessário uma etapa de pré-processamento da imagem em RGB para escala de cinza. A resposta do modelo consiste em quatro valores inteiros para cada região contendo uma face na imagem, (x, y, w, h). Onde x e y são as coordenadas do ponto superior esquerdo da região, e w e h são a largura e altura da região respectivamente. Um exemplo de detecção está representado na figura 9.

O método foi escolhido pelo baixo custo computacional necessário para realizar uma detecção. Isso se dá principalmente pela estrutura de cascata utilizada (Figura 10). Os classificadores intermediários levam em consideração apenas as regiões já verificadas pelos classificadores anteriores, o que faz com que os primeiros classificadores rejeitem um grande número regiões rapidamente.

Cada classificador no processamento utiliza uma máscara baseada em uma característica de *Haar* (Figura 11), onde a região em branco do filtro representa pesos positivos e as regiões em preto representam pesos negativos. O classificador realiza a soma da multiplicação ponto a ponto da máscara por uma região da imagem. O resultado dessa soma será então classificado por um limiar escolhido no processo de treinamento do modelo. Esta operação resume-se em identificar se a diferença de intensidade entre os pixeis da região branca com a região preta é

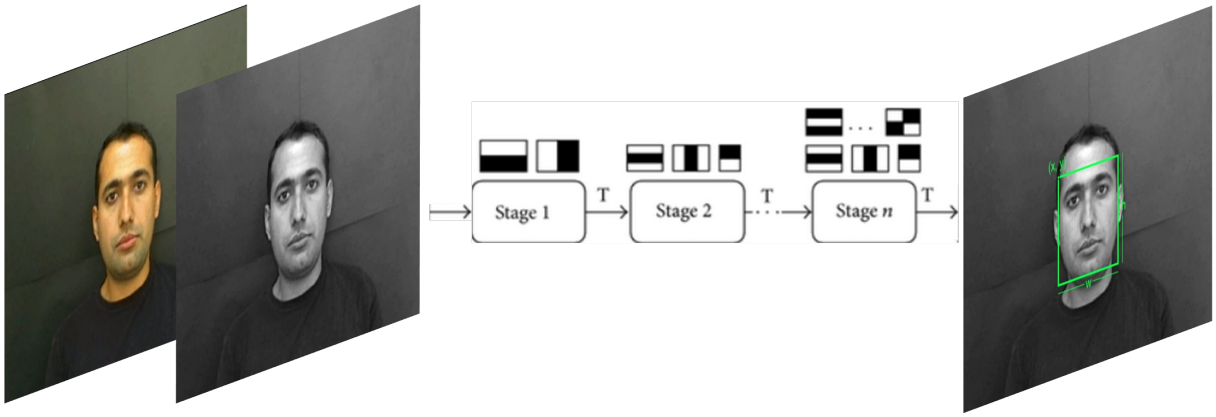

Figura 9 – Representação da etapa de detecção facial para um exemplo da base de dados BUH-MAP.

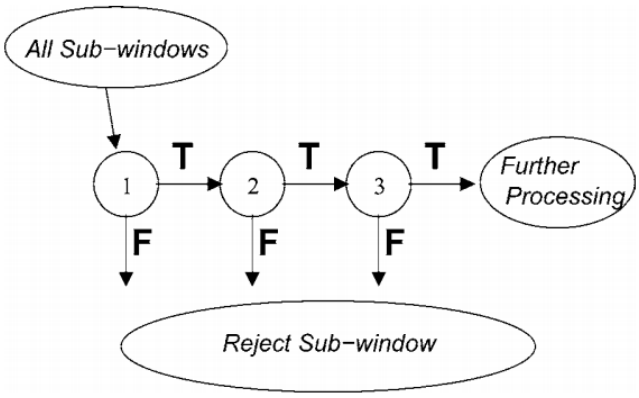

Figura 10 – Representação do processo de classificação em cascata, retirado de (VIOLA; JONES, 2004).

suficientemente grande para representar a característica buscada. Na figura 11 estão representados dois classificadores e suas respectivas máscaras, onde a primeira máscara refere-se a diferença de intensidade entre a região dos olhos e a região das bochechas superiores. Enquanto a segunda refere-se a diferença de intensidade nas regiões dos olhos com a intensidade através da ponta do nariz.

O modelo disponibilizado pela biblioteca foi desenvolvido pelos autores em (LIENHART; KURANOV; PISAREVSKY, 2003). Os classificadores foram treinados utilizando 5000 exemplos positivos (contendo pelo menos uma face) e 3000 exemplos negativos (não contendo nenhuma face), alcançando uma acurácia superior a 95%. Neste trabalho os autores propuseram uma série de novos classificadores obtidos através de rotações para as características baseadas em *Haar* utilizadas em (VIOLA; JONES, 2004). Utilizando as características rotacionadas, os autores reduziram em média 10% do número de falsos positivos identificados pelo modelo. Os filtros utilizados estão representadas na Figura 12 e foram divididos pelos respectivos padrões que são responsáveis por identificar. O primeiro grupo de filtros é responsável por reconhecer

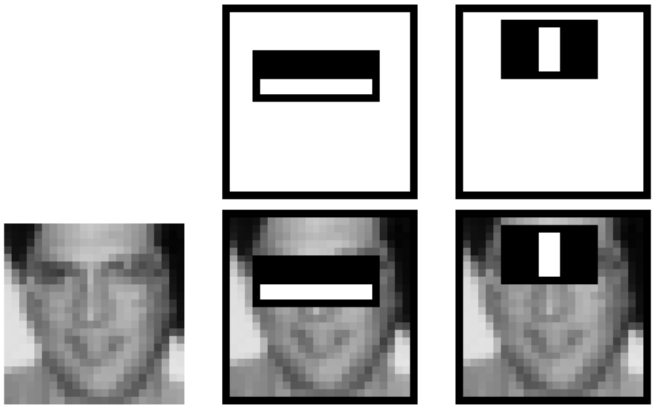

Figura 11 – Representação de dois classificadores de características de *Haar* mostrados na linha superior e depois sobrepostas em uma face de treinamento genérica na linha inferior, retirado de (VIOLA; JONES, 2004).

bordas na imagem, enquanto o segundo grupo é é capaz de reconhecer padrões lineares. O terceiro grupo deve identificar padrões circulares ou pontos. O quarto grupo é responsável por identificar padrões especiais de alguma forma relacionados na diagonal.

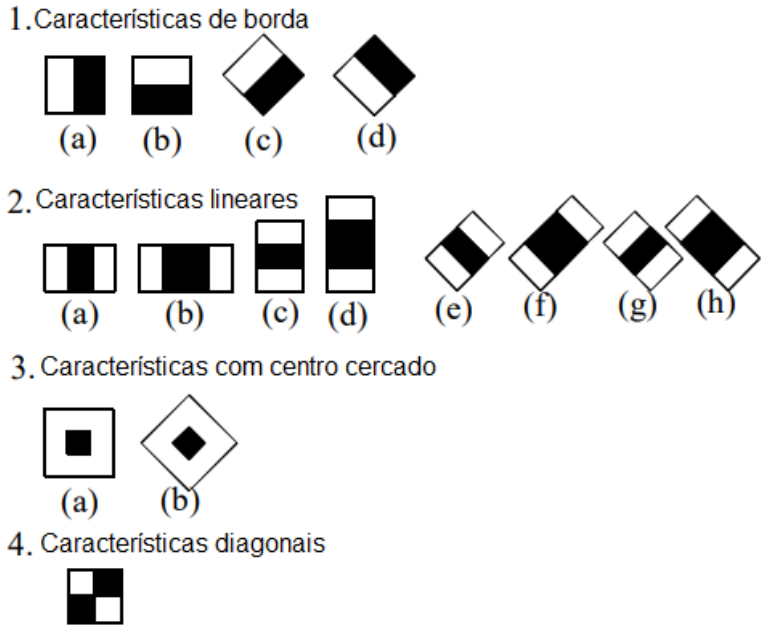

Figura 12 – Representação dos classificadores baseados em características de *Haar* e as suas respectivas rotações. A região em branco representa pesos positivos, enquanto a região em preto pesos negativos. Adaptado de (LIENHART; KURANOV; PISAREVSKY, 2003).

3.1.3 Detecção dos Pontos Fiduciais Faciais

Uma vez conhecida a região facial na imagem, é realizada a detecção dos pontos fiduciais faciais (PFF) que serão utilizados como referência para estimativa de posição angular da cabeça do usuário. A detecção destes pontos foi realizada através do modelo de regressão em cascata

proposto em (KAZEMI; SULLIVAN, 2014). O modelo utiliza um conjunto de árvores de regressão conectadas em cascata (Figura 13). A posição dos pontos iniciais é escolhida através da média desses pontos na base de dados utilizada no treinamento. O modelo recebe as posições iniciais e em cada processamento de uma árvore de regressão, é realizado um refinamento desses pontos para melhor ajustar a face da imagem. Desta forma uma árvore intermediária realiza o refinamento da estimativa da árvore anterior. Cada nó existente na árvore utiliza como parâmetro a diferença de intensidade entre dois PFF na imagem.

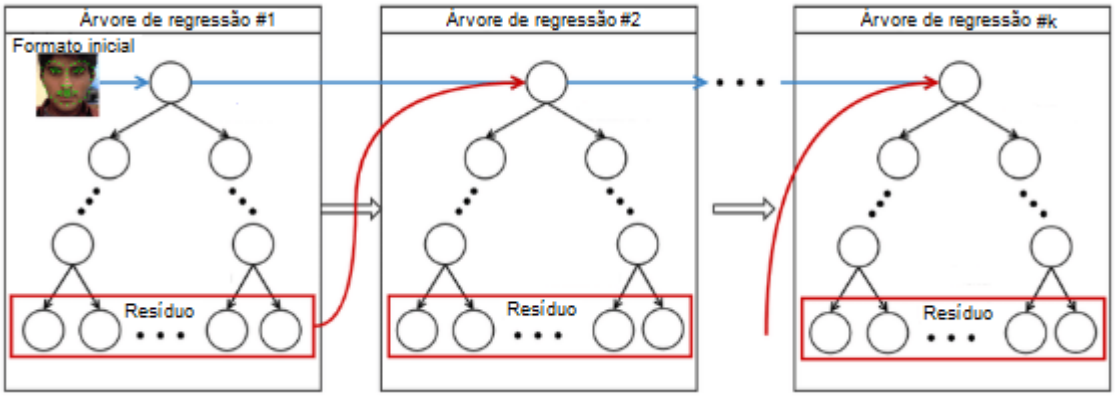

Figura 13 – Exemplificação do modelo em cascata de árvores de regressão (Ensemble Regression Tree), retirado de (KAZEMI; SULLIVAN, 2014).

Foi utilizado um modelo pré-treinado disponibilizado pela biblioteca Dlib, o qual é capaz de identificar as coordenadas x e y de 68 pontos representados na figura 14. O modelo recebe como entrada o frame e a região contendo a face detectada na etapa anterior, enquanto a saída consiste em uma lista de tamanho 2x68 contendo as coordenadas de cada ponto.

Figura 14 – Localização dos 68 pontos fiduciais detectados através do modelo, utilizando um frame de exemplo da base de dados BUHMAP.

3.1.4 Estimativa de Orientação Facial

A estimativa dos três ângulos de orientação v, w e u (*Pitch, Yaw, Roll*, representados na Figura 2) é realizada relacionando um modelo genérico tri-dimensional com a sua projeção no espaço bi-dimensional. O modelo antropomórfico genérico utilizado foi desenvolvido pelo *Institute of Systems and Robotic* da Universidade de Coimbra e está disponibilizado em `http://aifi.isr.uc.pt`. Esse modelo foi adquirido por uma varredura 3D frontal por laser de um modelo físico, selecionando os pontos 3D equivalentes do procedimento de anotação (Figura 15).

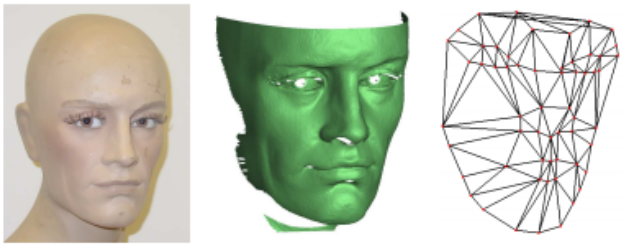

Figura 15 – Molde físico de um rosto genérico, varredura 3D do modelo físico, Modelo Antropomórfico 3D dos pontos selecionados, retirado (MARTINS; BATISTA, 2008).

Os PFF identificados na etapa anterior, serão interpretados como a projeção dos pontos tri-dimensionais do modelo no plano da imagem digital. Desta forma, a estimativa é realizada solucionando um problema PnP (Perspective-n-Point) ilustrado na figura 16. O problema consiste em estimar a perspectiva da câmera em relação ao objeto, a partir da imagem digital e dos parâmetros da captura.

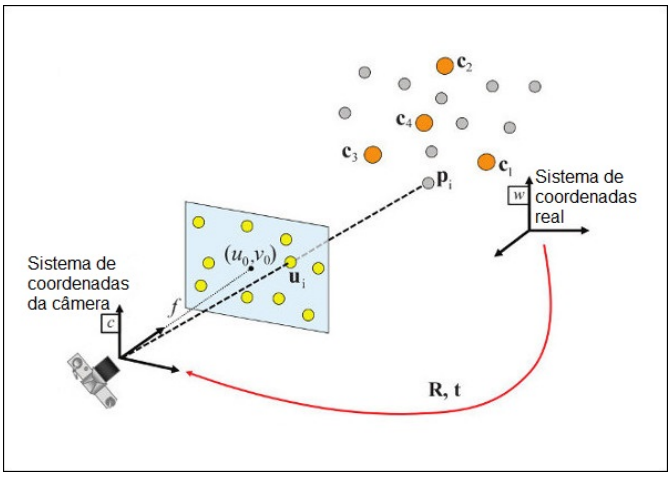

Figura 16 – Esquematização do problema PnP, extraído de (OpenCV, Open Source Computer Vision, 2019b).

As equações 3.1 e 3.2 descrevem a transformação entre as projeções m na imagem digital

e os pontos tri-dimensionais M no plano real.

$$sm = A[R|t]M \tag{3.1}$$

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.2}$$

Onde $M = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$ é o ponto real no espaço tri-dimensional, s é o fator de escala da imagem, f_x e f_y são as distâncias focais, (c_x, c_y) é o ponto central da imagem, a matriz R corresponde as rotações e t as translações da câmera nos eixos x, y, z. Através da equação, percebe-se que caso a imagem seja redimensionada por um fator s, todas as coordenadas também devem ser proporcionalmente redimensionadas por esse mesmo fator. Os parâmetros intrínsecos da câmera não dependem da cena em questão, desta forma, uma vez estimados podem ser reutilizados em trabalhos futuros. A biblioteca *OpenCV* disponibiliza a função *solvePnP* responsável por calcular a matriz de rotação R e translação t. Uma descrição completa do problema pode ser encontrada na documentação (OpenCV, Open Source Computer Vision, 2019a). Utilizando a função *decomposeProjectionMatrix*, é possível decompor a matriz de orientação $[R|t]$ nos ângulos de *Euler* (*Pitch, Yaw e Roll*).

3.2 Classificação de Gestos estáticos

Neste trabalho foram considerados 9 posições estáticas da cabeça que usuário pode utilizar para indicar um gesto estático. Desta forma, cada gesto corresponde às posições **Norte**, **Sul**, **Leste**, **Oeste**, **Nordeste**, **Noroeste**, **Suldeste**, **Suldoeste** e **Neutro**. A classificação desses gestos foi realizada através de uma árvore de decisão representada na Figura 17, que utiliza como parâmetro os limiares $[Y_{max}, Y_{min}, X_{max}, X_{min}]$ e como entrada os ângulos de v e w. Os limiares representam as rotações mínimas e máximas em cada eixo para a orientação deixar de ser considerada neutra. Por exemplo, a cabeça do usuário deverá estar orientada com um angulo de *Pitch* superior a Y_{max} para serem identificadas as posições **Norte**, **Noroeste** e **Nordeste**. Uma vez que o Módulo de Visão Computacional estimar os valores destes ângulos, será feita em seguida a classificação através de uma inferência por meio da árvore de Decisão.

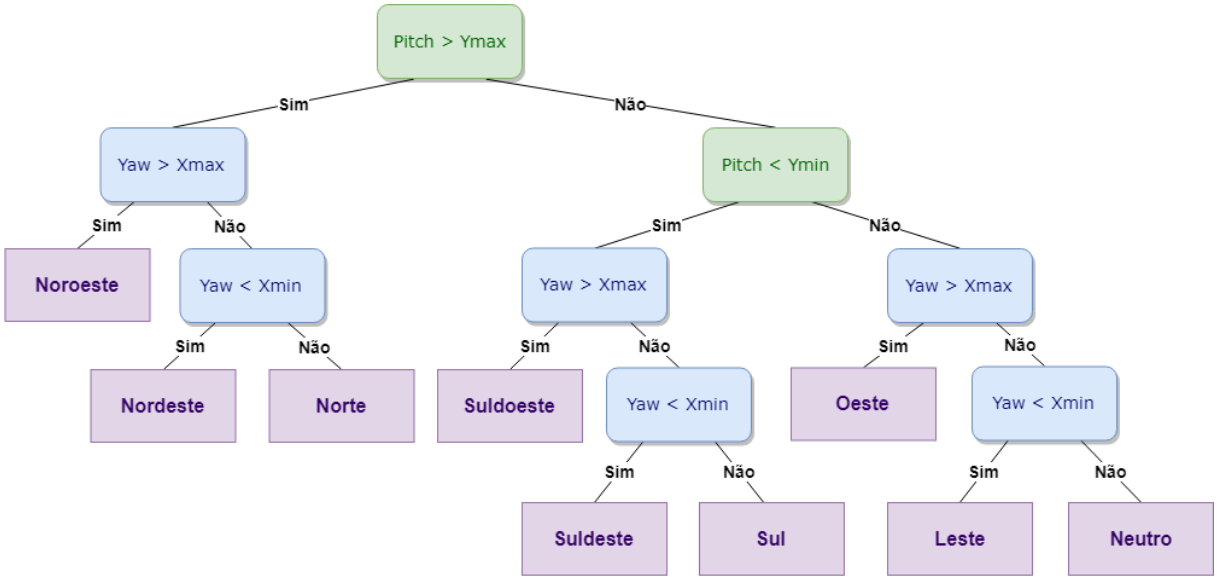

Figura 17 – Árvore de decisão para classificação dos gestos estáticos.

3.3 Módulo de Análise de Sinais

O Módulo é responsável por monitorar os ângulos estimados pelo módulo de Visão Computacional ao longo do tempo e reconhecer possíveis gestos dinâmicos realizados com a cabeça. O módulo conta com as etapas de construção de um série temporal, detecção de movimento e classificação dos gestos.

3.3.1 Construção da série temporal

A construção dessas séries é feita a partir da concatenação dos valores estimados para os ângulos de rotação ao longo do tempo. A orientação da cabeça é representada por três ângulos de rotação, por isso a série temporal conterá 3 canais respectivos a cada eixo de rotação. Cada ângulo sera inserido ao final da série em um dos três canais respectivos ao seu eixo. Cada vetor terá dimensão fixa igual a 50 posições, correspondendo a aproximadamente 4 segundos de execução. A atualização dos novos valores estimados, é feita através de uma rotação dos vetores em uma posição e substituição do último valor pelos novos, este processo está representado na figura 18.

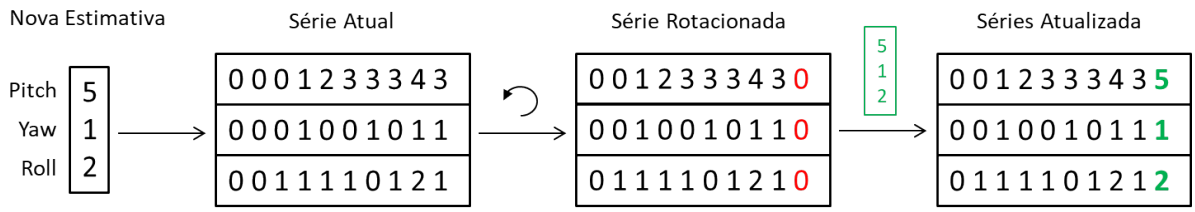

Figura 18 – Representação do processo de atualização das séries temporais.

3.3.2 Detecção de Movimentos

Para evitar o processamento desnecessários de dados irrelevantes ou ruídos, os sinais são analisados de forma a detectar a janela contendo movimentos significativos. Caso nenhum movimento seja identificado, não será necessário executar a etapa de classificação. Um movimento será considerado significativo a partir da magnitude da velocidade angular ($|\dot{\Theta}|$) de rotação nos três eixos. Para isso, esta magnitude é calculada através do módulo da derivada do sinal em cada um dos canais separadamente, como mostra a equação 3.3. A janela de um movimento será definida através dos pontos de início e fim do movimento. O início de um movimento será determinado quando a condição 3.6 for atendida, isto é, quando o módulo da velocidade for maior que o limiar para as próximas 4 estimativas seguidas. Enquanto o fim do movimento será identificado quando atendida a condição 3.5, ou seja, o módulo da derivada for menor que o limiar para as próximas 4 estimativas seguidas.

$$|\dot{\theta}_i| = \sqrt{(\theta_i - \theta_{i-1})^2} \quad , \quad para \quad i = [2, N] \tag{3.3}$$

$$\alpha_i = \begin{cases} 1, & |\dot{\theta}_i| >= t \\ 0, & |\dot{\theta}_i| < t \end{cases} \tag{3.4}$$

$$fim = i, \quad se, \quad \sum_{i}^{i+3} \alpha_i = 0 \tag{3.5}$$

$$inicio = i, \quad se, \quad \sum_{i}^{i+3} \alpha_i = 4 \tag{3.6}$$

A Figura 19 exemplifica o processo de detecção de movimento, utilizando como exemplo os três canais do sinal gerado a partir do movimento correspondente ao gesto "Não". As linhas pretas representam a trajetória da posição angular para as rotações u, v e w. Enquanto as curvas em laranjas correspondem ao módulo das velocidades angulares $\dot{u}$, $\dot{v}$ e $\dot{w}$. Os pontos verdes indicam os instantes em que a magnitude da velocidade angular ultrapassou o limiar escolhido t. Já os pontos em vermelho indicam os instantes em que a magnitude não ultrapassou o limiar.

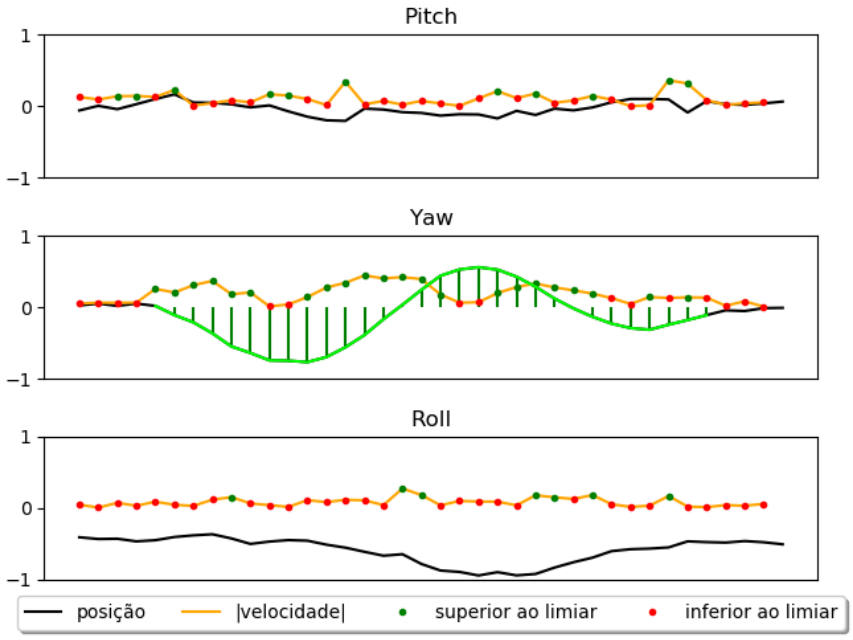

Figura 19 – Processo de detecção utilizando como o exemplo o sinal gerado pelo movimento correspondente ao gesto "Não", utilizando limiar= 0.134.

A escolha do limiar t foi realizada de forma manual e exploratória utilizando a base de dados BUHMAP. O valor considerado ótimo deve garantir que a maioria dos movimentos significantes sejam detectados, ou seja, uma maior taxa de revocação possível, mesmo assumindo uma certa taxa de falsos alarmes. Também foi levado em consideração o fato de que a janela de movimento deveria conter toda sua trajetória ao longo do tempo, evitando a perda de informações. Desta forma, o valor encontrado foi de $t = 0.134$, obtendo uma taxa de 100% de acerto para verdadeiros positivos mantendo uma taxa de 32% de falsos positivos.

3.3.3 Pré-processamento de Dados

Antes de serem submetidos à etapa de classificação, os sinais de entrada são pré-processados. Esta etapa consiste na transformação dos dados de forma a aprimorar o desempenho do classificador. O primeiro passo foi calcular uma aproximação da derivada de cada um dos canais através da diferenciação ponto a ponto, desta forma o sinal resultante será centralizado no valor de amplitude 0 e representará as velocidades angulares $\dot{v}$, $\dot{w}$ e $\dot{u}$, para as rotações de *Pitch, Yaw e Roll*. O segundo passo consiste na normalização da amplitude dos sinais resultantes entre os valores -1 e +1.

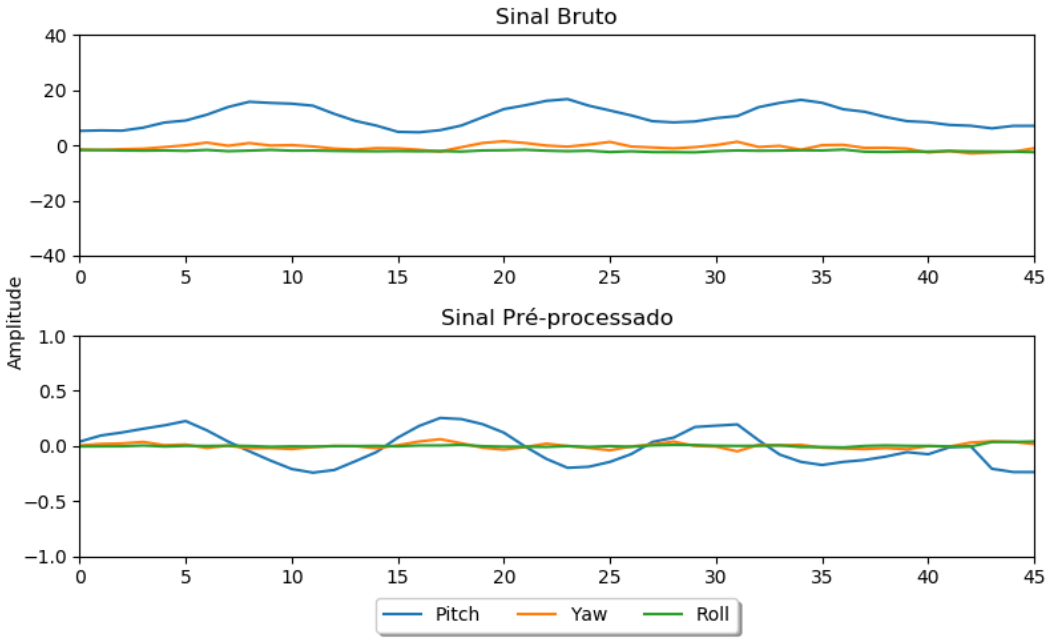

Figura 20 – Comparação entre o sinal bruto e pré-processado.

3.3.4 Classificação de Sinais

O processo de classificação de sinais será responsável por atribuir uma categoria (Classe) aos dados de entrada. As classes a serem consideradas serão os gestos dinâmicos "Sim"e "Não"e a ausência de gestos como "Neutro", representados na Figura 21. O gesto "Sim"corresponde ao movimento para cima e para baixo da cabeça, enquanto o gesto "Não"corresponde ao movimento de um lado para o outro.

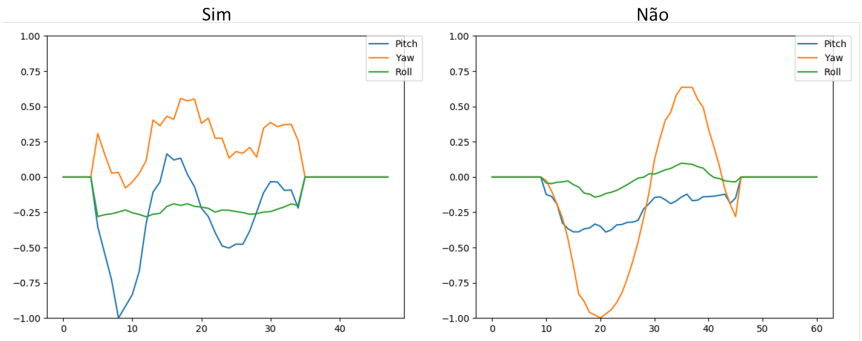

Figura 21 – Exemplos de sinais de entrada para a etapa de classificação. O gráfico da esquerda corresponde ao gesto *"Sim"*, enquanto o gráfico da direita corresponde ao *"Não"*.

A classificação foi realizada através de uma CNN unidimensional. Esta rede diferencia-se das CNN tradicionais, pois a operação de convolução será feita em apenas uma dimensão (Duração do Sinal). Desta forma a máscara se deslocará em um único eixo, como mostrado na figura 22.

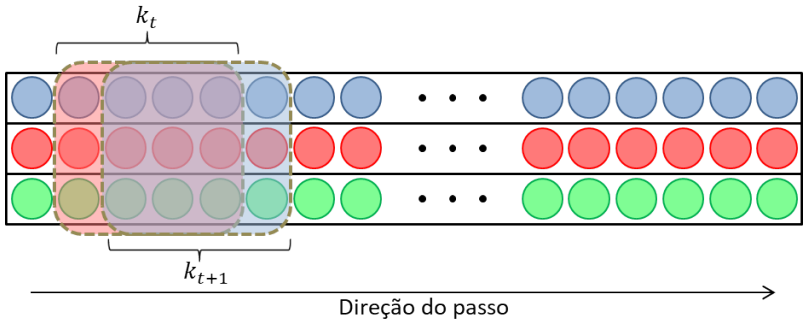

Figura 22 – Demonstração da operação de Convolução unidimensional, utilizada pela 1D-CNN.

A arquitetura escolhida para a CNN possui 10 camadas e um total de 237827 parâmetros ajustáveis, como apresentado na Figura 23. O vetor de características possui 50 posições para cada um os 3 canais. A rede pode ser dividida em duas regiões, onde a primeira desempenha o papel de extrair as características contidas nos dados de entrada, enquanto a segunda desempenha o papel de classificar essas características.

A primeira região da rede conta com uma sequencia de camadas de convolução unidimensional utilizando a função de ativação ReLU, intercaladas com camadas de *maxpooling*. Todas as camadas convolucionais utilizam a estratégia de *padding* para manter o tamanho do mapa de ativação igual ao tamanho da entrada. Além disso, as máscaras utilizadas nessas camadas possuem tamanho 3x3. As camadas de redução de dados utilizam um fator de deslocamento igual a 2, reduzindo pela metade o tamanho do mapa de ativação.

A segunda região da rede concatena os valores da saída da camada Conv1D-4, gerando um vetor de uma dimensão contendo 1536 valores. O vetor resultante é então classificado por três camadas totalmente conectadas, assim como no MLP. As camadas FC1 e FC2 utilizam ReLU como função de ativação, enquanto a camada FC3 utiliza a operação de *Softmax*. A arquitetura resultante da CNN está representada na Figura 23.

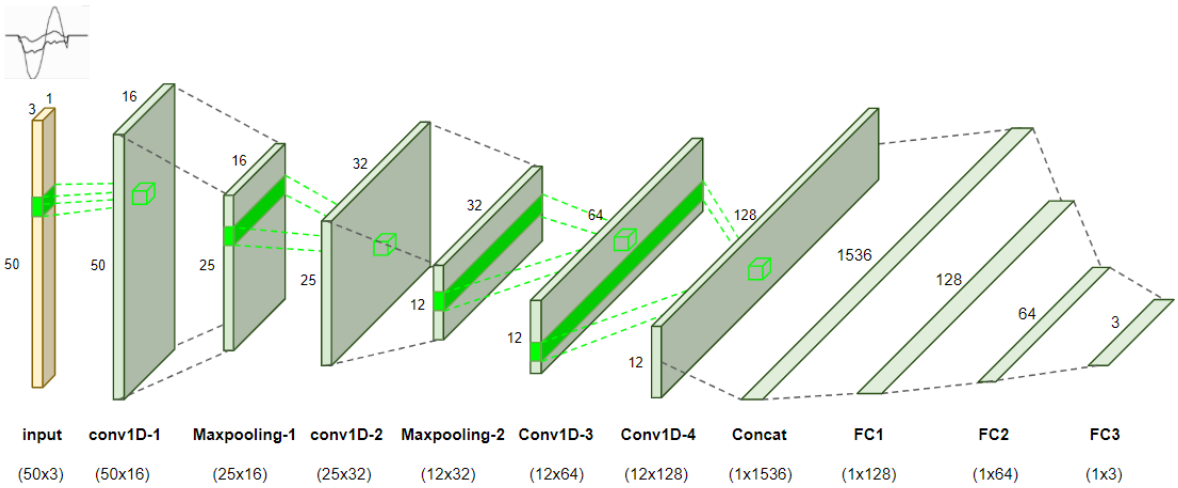

Figura 23 – Representação da arquitetura utilizada para CNN.

3.4 Comandos

O sistema final utiliza o reconhecimento de gestos apresentado anteriormente como interface para controlar a navegação do usuário pelo Facebook. Esta aplicação foi escolhida por se tratar de uma operação simples e cotidiana a qual a limitação dos movimentos tem alto impacto.

Como gestos estáticos, é proposto a identificação as posições *Norte*, *Sul*, *Leste*, *Oeste* e *Noroeste* indicadas com a cabeça. Enquanto os gestos dinâmicos a serem reconhecidos serão os movimentos, *Sim* (movimento para cima e para baixo) e *Não* (movimentos para um lado e para o outro) com a cabeça. Cada um dos gestos indicados resultará em uma ação de controle conforme relacionado na tabela abaixo.

Tabela 1 – Lista dos gestos identificados pelo sistemas e suas respectivas ações.

Tipo	Gesto	Ação
Estático	Sul	Mover o feed para o próximo post
	Norte	Mover o feed para o post anterior
	Leste	Expandir o post selecionado
	Oeste	Sair do post expandido
	Noroeste	Curtir post atual
Dinâmico	Sim	Confirmar caixa de diálogo
	Não	Cancelar caixa de diálogo

Para interação com o aplicativo Facebook, o sistema utiliza de atalhos do teclado para realizar cada tarefa, uma vez que o aplicativo possui uma interface de acessibilidade (Figura 24) a qual compreende alguns atalhos de teclado como ações na página. Desta forma, o sistema será responsável por traduzir os gestos realizados com a cabeça em atalhos do teclado.

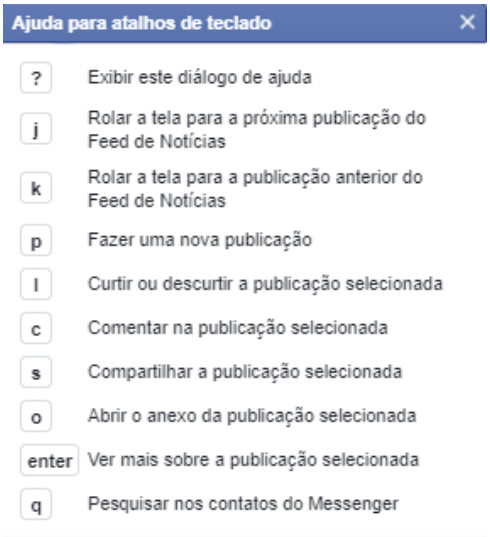

Figura 24 – Tabela de atalhos de teclado para acessibilidade no Facebook.

4 EXPERIMENTOS

Com intuito de avaliar o desempenho da abordagem de classificação de gestos dinâmicos, bem como a extração de características proposta, foram realizados uma série de experimentos utilizando uma base de dados retirada da literatura. A base escolhida foi desenvolvida em ambiente controlado para aquisição dos dados, servindo como teste inicial de desempenho. Os experimentos e resultados, bem como a base de dados, estão detalhados neste capítulo.

4.1 Base de dados

Para os experimentos apresentados em seguida, foi utilizada a base de dados BUHMAP. A base contem gestos livres do uso das mãos utilizados na língua de sinais da Turquia. A base de dados foi construída através do vídeo de 11 pessoas (6 mulheres, 5 homens) realizando 8 gestos faciais. Cada pessoa executou 5 vezes cada um dos gestos, gerando um espaço amostral de 40 vídeos por pessoa e totalizando 440 vídeos. Cada um dos vídeos gerados tem de 1 a 2 segundos de duração e para a aquisição das imagens, foi utilizada uma webcam Philips SPC900NC com uma resolução de 640x480 pixeis a uma taxa de 30 FPS. A descrição de cada um dos gestos e suas respectivas classes associadas estão apresentadas na tabela 2.

Tabela 2 – Descrição dos gestos realizados nos vídeos da base de dados BUHMAP.

Gesto	Classe	Descrição
Neutro	0	Estado neutro da face durante todo o vídeo (ausência de movimento)
Não	1	Mover a cabeça de um lado para o outro (Movimento de negação)
Olhar para cima	2	Movimentar a cabeça de forma a olhar para cima e retornar para posição neutra
Aproximar da câmera	3	Inclinar o corpo para frente, aproximando a face da câmera
Tristeza	4	Movimentar os lábios e as sobrancelhas para baixo e levemente mover a cabeça para baixo
Sim	5	Mover a cabeça para cima e para baixo (Movimento de aceitação)
Felicidade	6	Movimentar os lábios para cima e sorrir
Sim + Felicidade	7	Execução dos gestos 5 e 6 ao mesmo tempo

4.2 Aumento de dados

A base de dados escolhida possui um número de exemplos relativamente baixo considerando o seu uso para ajustar uma técnica de *DL*. Técnicas como uma CNN possuem um grande número de parâmetros, permitindo aprender com facilidade padrões complexos nas amostras. Por isso, uma CNN apresenta facilidade em ocorrer *Sobreajuste* para a base de dados quando o número de exemplos são baixos. Isto é, o modelo se tornará especialista nos exemplos utilizados, apresentando um baixo desempenho quando utilizado em um ambiente real. Por isso, para evitar o *Sobreajuste* e tornar o modelo generalista, foram utilizadas uma série de técnicas de aumento de dados, que estão detalhadas abaixo.

4.2.1 Inversão

A primeira técnica aplicada é a técnica de inversão, que consiste na multiplicação de todos os valores do sinal por -1. Desta forma, o sinal resultante será a inversão do sinal original no eixo x. A inversão foi aplicada para todos os exemplos da base de dados duplicando o número de exemplos de 440 para 880. Vale ressaltar que os gestos devem ser simétricos ao eixo x para que se possa utilizar esta técnica.

4.2.2 Compressão

A segunda técnica aplicada consiste na compressão do sinal original tornando-o mais breve. Isto é feito removendo aleatoriamente valores intermediários e preenchendo com valores nulos as bordas, de forma a manter o mesmo tamanho do vetor inicial. A compressão também foi realizada para todos os exemplos da base de dados já aplicada o aumento de dados por inversão. Ao final desta etapa o número de exemplos passou de 880 para 1760.

4.2.3 Translação

A última técnica consiste em deslocar o sinal em relação a sua posição original. Neste trabalho foram realizados 6 diferentes translações, considerando um tamanho de deslocamento de 2, 4 e 8 unidades para direita e para a esquerda. Desta forma, para cada amostra na base de dados, 6 novas amostras foram geradas. O número final de exemplos alcançados subiu para 12320, o que compõem uma amostragem 28 vezes maior que a amostragem inicial.

4.3 Desenvolvimento da CNN

Utilizando a base de dados BUHMAP, foi desenvolvida uma CNN unidimensional para a classificação dos gestos dinâmicos realizados com a cabeça. Cada um dos vídeos foi processado pelo módulo de visão computacional gerando os sinais de entrada como descrito na seção 3.3.1. Além disso, foram realizadas as técnicas de aumento de dados, *translação*, *compressão* e *inversão* explicadas na seção anterior. Com o aumento dos dados, as amostras de treino passaram a contar com 7840 exemplos, enquanto as amostras de validação contam com 4480 exemplos.

A divisão escolhida para a base de dados foi de 63% (todos os exemplos de 7 indivíduos) para etapa de treino e 36% (todos os exemplos de 4 indivíduos) para a etapa de validação. Esse método de divisão foi escolhido para balancear o número de exemplos por classe e evitar exemplos executados por uma mesma pessoa em ambas etapas.

Os gestos correspondentes às classes 6 e 4 são de expressão facial, desta forma, não é executado nenhum movimento com a cabeça durante o vídeo. Além disso, o gesto correspondente à classe 8 é composto pelo mesmo movimento realizado no gesto correspondente à classe 7, adicionando apenas uma expressão facial de felicidade. Por isso, foram realizados três diferentes ensaios utilizando diferentes amostragens para a base de dados, considerando diferentes classes

e mantendo a proporção de exemplos entre treino e validação. Cada uma das amostragens está descrita na tabela 3.

Tabela 3 – Descrição das três amostragens realizadas na base de dados com as técnicas de aumento de dados.

Amostragem		Total de exemplos	Exemplos por Classe	Exemplos por indivíduo
D1	Treino	7840	980	1120
	Validação	4480	560	1120
D2	Treino	4900	980	700
	Validação	2800	560	700
D3	Treino	2940	980	420
	Validação	1680	560	420

A amostragem *D1* consiste na base de dados original considerando todas as classes. A amostragem *D2* consiste em todos os exemplos considerando apenas as classes $[0, 1, 2, 3, 5]$. A amostragem *D3* consiste em todos os exemplos considerando apenas classes $[0, 1, 5]$.

4.3.1 Ensaio 1

No primeiro momento, o modelo foi treinado e avaliado a partir da amostragem *D1*. O objetivo deste ensaio é obter o resultado para comparação com os próximos ensaios utilizando as diferentes amostragens propostas para avaliar o método. Conforme mostrado na tabela abaixo, o modelo não alcança um desempenho significativo. Um possível motivo pelo baixo desempenho é a confusão entre as classes que realizam o mesmo movimento durante o gesto (5 e 7) ou que nenhum movimento (0, 3, 4 e 6). Isto pode ser concluído através dos resultados isolados das classes 1 e 2 em que o modelo alcança os valores de F1, equivalentes a 97% e 74% respectivamente.

Tabela 4 – Resultado para o Ensaio 1.

Classe	Precisão	Revocação	F1	Amostras
0	0.74	0.74	0.74	560
1	0.99	0.94	0.97	560
2	0.64	0.86	0.74	560
3	0.44	0.29	0.35	560
4	0.49	0.27	0.35	560
5	0.37	0.40	0.38	560
6	0.61	0.52	0.56	560
7	0.50	0.79	0.61	560
			Acurácia	0.60

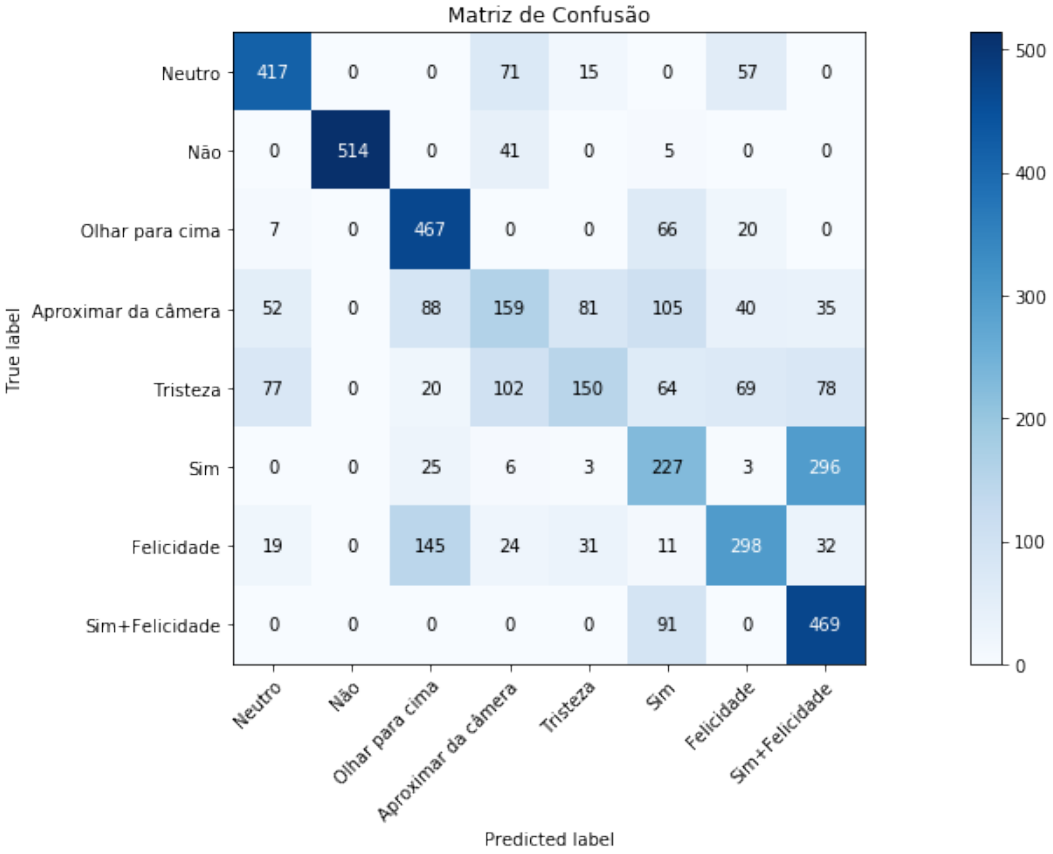

Figura 25 – Matriz de confusão para o primeiro ensaio considerando todas as classes da base de dados

4.3.2 Ensaio 2

Para confirmar a suposição levantada pelo resultado do Ensaio 1, foi realizado um segundo ensaio excluindo as classes consideradas ambíguas em relação ao movimento executado. Desta forma, o modelo foi treinado utilizando a amostragem *D2* contendo os exemplos apenas para as classes $[0, 1, 2, 3, 5]$. Os resultados apresentados na tabela abaixo comprovam uma melhora significativa no desempenho do modelo, uma vez que, a acurácia média do modelo subiu de 60% para 88% quando consideradas apenas as classes com movimentos não ambíguos.

Tabela 5 – Resultado para o Ensaio 2.

Classe	Precisão	Revocação	F1	Amostras
0	0.83	0.86	0.85	560
1	1.00	0.96	0.98	560
2	0.82	0.99	0.90	560
3	0.79	0.65	0.71	560
5	0.96	0.94	0.95	560
			Acurácia	0.88

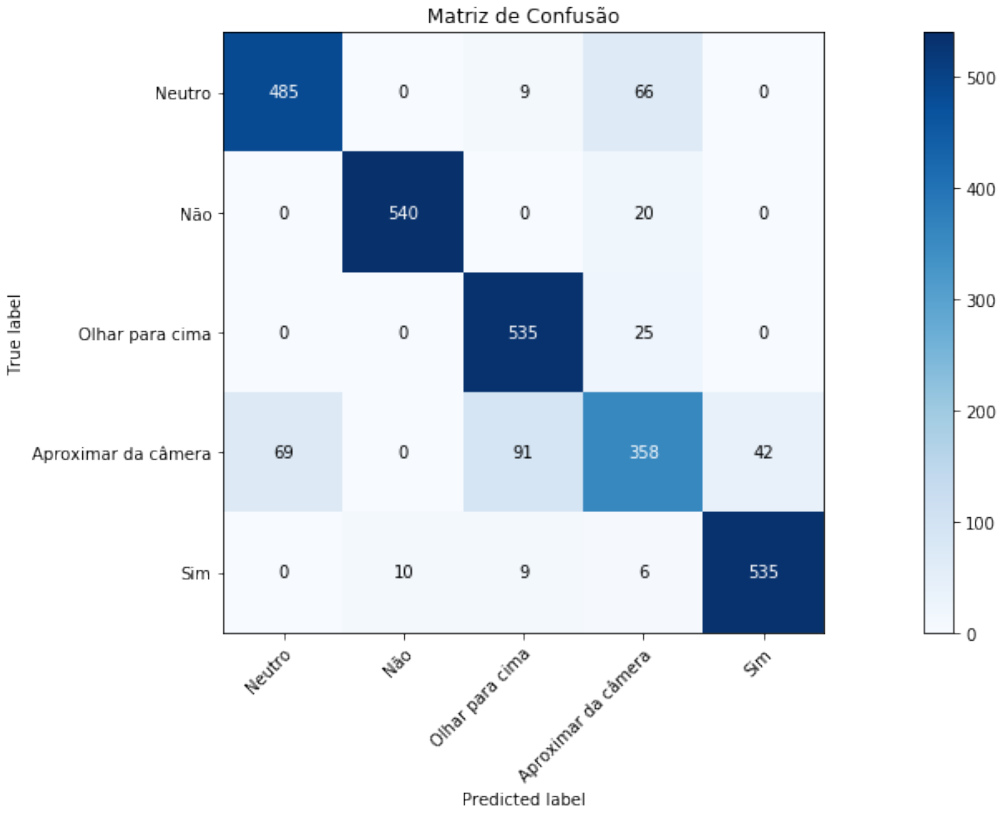

Figura 26 – Matriz de confusão para o segundo ensaio, considerando 5 classes da base de dados.

4.3.3 Ensaio 3

Para avaliar o desempenho do modelo em um contexto mais próximo da aplicação final, neste ensaio foram consideradas apenas as classe $[0, 1, 5]$ que, equivalem aos gestos de **Sim** e **Não** utilizados no sistema final e a ausência de gestos. O desempenho do modelo neste contexto subiu para 98% de acurácia média, o que confirma a eficiência do método em classificar os gestos a serem utilizados para um ambiente controlado.

Tabela 6 – Resultado para o Ensaio 3.

Classe	Precisão	Revocação	F1	Amostras
0	0.98	0.98	0.98	560
1	1.00	0.98	0.99	560
5	0.96	0.98	0.97	560
			Acurácia	0.98

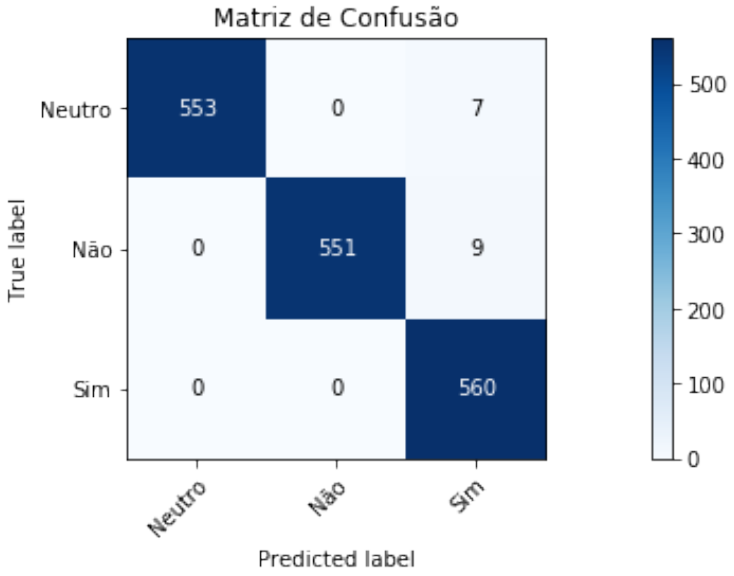

Figura 27 – Matriz de confusão para o terceiro ensaio, considerando 3 classes da base de dados.

4.4 Resultados

Analisando a matriz de confusão do primeiro ensaio, onde o modelo apresentou pior desempenho, foi possível levantar algumas suposições relacionadas aos resultados para cada classe. Os melhores resultados correspondiam às classes "Não"e "Sim+Felicidade", o que leva a acreditar que a técnica alcançará seu objetivo no sistema final. Além disso, os piores resultados estão relacionados a confusão entre classes consideradas ambíguas em relação ao movimento. Por exemplo, a classe "Sim"foi confundida na maior parte das vezes pela classe "Sim+Felicidade", e as classes "Tristeza"e "Felicidade"foram mais confundidas com a classe "Neutro". Isto se deu ao fato de que, como o método concentra-se em distinguir apenas a trajetória da cabeça, informações como a expressão facial não são percebidas.

Esta suposição foi confirmada com a melhora significativa de desempenho no ensaio 2, elevando de 60% à 88% a acurácia média, onde foram removidas essas classes ambíguas de movimento. A partir da Tabela 5 nota-se também que os melhores resultados correspondem as classes "Não"e "Sim", alcançando 98% e 95% de F1 respectivamente. Como essas classes correspondem aos gestos a serem considerados na interface proposta, foi realizado um terceiro ensaio desenvolvendo o modelo especialista para essa tarefa. O resultado deste último ensaio confirma a capacidade promissora em distinguir os gestos *"Sim"* e *"Não"*, alcançando uma acurácia média de 98%.

Os experimentos realizados mostram que o método abordado neste trabalho apresentou um bom desempenho em discriminar os movimentos realizados em ambiente controlado. Estes resultados servem como primeiro passo para o desenvolvimento de uma plataforma robusta o suficiente para ser distribuída, uma vez que, o desempenho promissor confirma a possibilidade

de interação com o computador utilizando técnicas de visão computacional combinadas com aprendizado de máquina.

CONCLUSÃO

O número de pesquisas envolvendo a aplicação de técnicas de Visão Computacional apresentou um aumento relevante com o passar dos anos, especialmente a partir dos anos 1980 e 2000. Esse aumento pode ser considerado um reflexo de uma série de trabalhos essenciais para o desenvolvimento dessa área, apresentados nesse período. Essas técnicas alcançaram desempenho relevante a partir do momento em que foi possível "aprender"automaticamente a partir de exemplos (Aprendizado de Máquina), ter acesso a um grande volume de amostras e "imitar"o funcionamento do cérebro humano (Redes Neurais).

As tecnologias atuais alcançam desempenhos bastantes significativos em realizar tarefas como a detecção facial, detecção de expressão do usuário, estimativa de orientação do rosto e classificação de gestos. Unindo algumas destas técnicas foi possível identificar os gestos dinâmicos "Sim"e "Não"na base de dados BUHMAP, alcançando uma acurácia de 98%, o que possibilita o uso de gestos naturais do ser humano como forma de interação com o computador. Além disso, foi possível também identificar posições específicas da cabeça do usuário como gestos estáticos.

Neste trabalho, diversas técnicas de VC e reconhecimento de padrões para criar uma interface de controle homem-máquina independente do uso das mãos foram combinadas. O protótipo desenvolvido permite ao usuário navegar pelo Feed de Notícias do Facebook realizando movimentos com a cabeça. Os resultados alcançados nos experimentos confirmam a capacidade desta área de pesquisa para inovar o acesso ao mundo digital, permitindo contornar possíveis dificuldades de pessoas com limitações motoras. O protótipo desenvolvido abre portas para o desenvolvimento de uma interface de acessibilidade mais robusta, possível de ser distribuída para uso pessoal.

Como trabalhos futuros, é proposto uma avaliação do desempenho do sistema em um ambiente não controlado e um estudo de caso envolvendo um usuário com algum grau de limitação. Como melhoria no software é proposto um modo de customização dos comandos, o que possibilitaria ao usuário escolher quais ações corresponderiam a cada gesto. Além disso, para facilitar a execução dos gestos é proposta uma etapa de calibração dos movimentos ao gosto do usuário. Por fim, existe também a intenção de explorar a inserção de novos recursos, como o reconhecimento de fala.

REFERÊNCIAS

AKAKIN, H. Ç.; SANKUR, B. Robust classification of face and head gestures in video. *Image and Vision Computing*, 2011. Elsevier, v. 29, n. 7, p. 470–483, 2011. Citado na página 21.

ALTHOFF, F. et al. Robust multimodal hand-and head gesture recognition for controlling automotive infotainment systems. *VDI BERICHTE*, 2005. VDI; 1999, v. 1919, p. 187, 2005. Citado na página 23.

ARI, I.; UYAR, A.; AKARUN, L. Facial feature tracking and expression recognition for sign language. In: IEEE. *2008 23rd International Symposium on Computer and Information Sciences*. [S.l.], 2008. p. 1–6. Citado 2 vezes nas páginas 22 e 27.

BÄCHLIN, M. et al. A wearable system to assist walking of parkinson s disease patients. *Methods of information in medicine*, 2010. Schattauer GmbH, v. 49, n. 01, p. 88–95, 2010. Citado na página 26.

BETKE, M.; GIPS, J.; FLEMING, P. The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities. *IEEE Transactions on neural systems and Rehabilitation Engineering*, 2002. IEEE, v. 10, n. 1, p. 1–10, 2002. Citado na página 26.

BRADSKI, G.; KAEHLER, A. *Learning OpenCV: Computer vision with the OpenCV library*. [S.l.]: "O'Reilly Media, Inc.", 2008. Citado na página 15.

COOK, A. M.; POLGAR, J. M. *Assistive Technologies-E-Book: Principles and Practice*. [S.l.]: Elsevier Health Sciences, 2014. Citado na página 26.

CROWLEY, J. et al. Finger tracking as an input device for augmented reality. In: *International Workshop on Gesture and Face Recognition*. [S.l.: s.n.], 1995. p. 195–200. Citado na página 19.

CUI, Z.; CHEN, W.; CHEN, Y. Multi-scale convolutional neural networks for time series classification. *arXiv preprint arXiv:1603.06995*, 2016. 2016. Citado na página 24.

DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. *2009 IEEE conference on computer vision and pattern recognition*. [S.l.], 2009. p. 248–255. Citado 2 vezes nas páginas 16 e 26.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern classification*. [S.l.]: John Wiley & Sons, 2012. Citado 2 vezes nas páginas 8 e 22.

DUMOULIN, V.; VISIN, F. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016. 2016. Citado 2 vezes nas páginas 8 e 25.

ERDEM, U. M.; SCLAROFF, S. Automatic detection of relevant head gestures in american sign language communication. In: IEEE. *Object recognition supported by user interaction for service robots*. [S.l.], 2002. v. 1, p. 460–463. Citado 2 vezes nas páginas 8 e 23.

ESTATÍSTICA, I. B. de Geografia e. *Pesquisa Nacional de Saúde 2013: ciclos de vida: Brasil e grandes regiões*. [S.l.]: IBGE Rio de Janeiro, 2015. Citado na página 17.

FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 1980. Springer, v. 36, n. 4, p. 193–202, 1980. Citado na página 16.

HJELMÅS, E.; LOW, B. K. Face detection: A survey. *Computer vision and image understanding*, 2001. Elsevier, v. 83, n. 3, p. 236–274, 2001. Citado na página 20.

HSU, R.-L.; ABDEL-MOTTALEB, M.; JAIN, A. K. Face detection in color images. *IEEE transactions on pattern analysis and machine intelligence*, 2002. IEEE, v. 24, n. 5, p. 696–706, 2002. Citado na página 20.

HUANG, J.; SHAO, X.; WECHSLER, H. Face pose discrimination using support vector machines (svm). In: IEEE. *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No. 98EX170)*. [S.l.], 1998. v. 1, p. 154–156. Citado na página 21.

JIA, P. et al. Head gesture recognition for hands-free control of an intelligent wheelchair. *Industrial Robot: An International Journal*, 2007. Emerald Group Publishing Limited, v. 34, n. 1, p. 60–68, 2007. Citado na página 26.

JIANG, H.; LEARNED-MILLER, E. Face detection with the faster r-cnn. In: IEEE. *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. [S.l.], 2017. p. 650–657. Citado na página 20.

KATSIKITIS, M. *The human face: measurement and meaning*. [S.l.]: Springer Science & Business Media, 2003. Citado na página 21.

KAZEMI, V.; SULLIVAN, J. One millisecond face alignment with an ensemble of regression trees. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 1867–1874. Citado 3 vezes nas páginas 8, 22 e 32.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2012. p. 1097–1105. Citado 2 vezes nas páginas 16 e 26.

LECUN, Y.; BENGIO, Y. et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 1995. v. 3361, n. 10, p. 1995, 1995. Citado 2 vezes nas páginas 8 e 24.

LECUN, Y. et al. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1989. MIT Press, v. 1, n. 4, p. 541–551, 1989. Citado na página 15.

LEE, H. et al. Unsupervised feature learning for audio classification using convolutional deep belief networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2009. p. 1096–1104. Citado na página 26.

LIENHART, R.; KURANOV, A.; PISAREVSKY, V. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In: SPRINGER. *Joint Pattern Recognition Symposium*. [S.l.], 2003. p. 297–304. Citado 3 vezes nas páginas 8, 30 e 31.

LIU, W. et al. Ssd: Single shot multibox detector. In: SPRINGER. *European conference on computer vision*. [S.l.], 2016. p. 21–37. Citado na página 26.

LIU, W. et al. A survey of deep neural network architectures and their applications. *Neurocomputing*, 2017. Elsevier, v. 234, p. 11–26, 2017. Citado na página 16.

MADHURI, Y.; ANITHA, G.; ANBURAJAN, M. Vision-based sign language translation device. In: IEEE. *2013 International Conference on Information Communication and Embedded Systems (ICICES)*. [S.l.], 2013. p. 565–568. Citado na página 27.

MARTINS, P.; BATISTA, J. Accurate single view model-based head pose estimation. In: IEEE. *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*. [S.l.], 2008. p. 1–6. Citado 2 vezes nas páginas 8 e 33.

MITRA, S.; ACHARYA, T. Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2007. IEEE, v. 37, n. 3, p. 311–324, 2007. Citado na página 19.

MURPHY-CHUTORIAN, E.; TRIVEDI, M. M. Head pose estimation in computer vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2008. IEEE, v. 31, n. 4, p. 607–626, 2008. Citado 2 vezes nas páginas 8 e 21.

NISHIKAWA, A. et al. Face mouse: A novel human-machine interface for controlling the position of a laparoscope. *IEEE Transactions on Robotics and Automation*, 2003. IEEE, v. 19, n. 5, p. 825–841, 2003. Citado na página 19.

NIYOGI, S.; FREEMAN, W. T. Example-based head tracking. In: IEEE. *Proceedings of the second international conference on automatic face and gesture recognition*. [S.l.], 1996. p. 374–378. Citado na página 21.

OHAYON, S.; RIVLIN, E. Robust 3d head tracking using camera pose estimation. In: IEEE. *18th International Conference on Pattern Recognition (ICPR'06)*. [S.l.], 2006. v. 1, p. 1063–1066. Citado na página 21.

OpenCV, Open Source Computer Vision. *Camera Calibration and 3D Reconstruction*. 2019. [Online; accessed June 15, 2019]. Disponível em: <https://docs.opencv.org/2.4-/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html>. Citado na página 34.

OpenCV, Open Source Computer Vision. *Real Time pose estimation of a textured object*. 2019. [Online; accessed April 29, 2019]. Disponível em: <https://docs.opencv.org/3.4/dc/d2c-/tutorial_real_time_pose.html>. Citado 2 vezes nas páginas 8 e 33.

OSADCHY, M.; CUN, Y. L.; MILLER, M. L. Synergistic face detection and pose estimation with energy-based models. *Journal of Machine Learning Research*, 2007. v. 8, n. May, p. 1197–1215, 2007. Citado na página 21.

QUEK, F. K. Unencumbered gestural interaction. *IEEE multimedia*, 1996. IEEE, v. 3, n. 4, p. 36–47, 1996. Citado na página 19.

TAIGMAN, Y. et al. Deepface: Closing the gap to human-level performance in face verification. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 1701–1708. Citado na página 26.

VIOLA, P.; JONES, M. J. Robust real-time face detection. *International journal of computer vision*, 2004. Springer, v. 57, n. 2, p. 137–154, 2004. Citado 6 vezes nas páginas 8, 16, 20, 29, 30 e 31.

WU, Y.; HUANG, T. S. Vision-based gesture recognition: A review. In: SPRINGER. *International Gesture Workshop*. [S.l.], 1999. p. 103–115. Citado na página 19.

ZELLER, M. et al. A visual computing environment for very large scale biomolecular modeling. In: IEEE. *Proceedings IEEE International Conference on Application-Specific Systems, Architectures and Processors*. [S.l.], 1997. p. 3–12. Citado na página 19.

ZHANG, C.; ZHANG, Z. A survey of recent advances in face detection. 2010. 2010. Citado na página 20.

ZHENG, Y. et al. Time series classification using multi-channels deep convolutional neural networks. In: SPRINGER. *International Conference on Web-Age Information Management*. [S.l.], 2014. p. 298–310. Citado na página 26.

ZHU, Z. et al. *Wearable navigation assistance for the vision-impaired*. [S.l.]: Google Patents, 2014. US Patent App. 14/141,742. Citado na página 26.