

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE
MINAS GERAIS - *CAMPUS* IBIRITÉ
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Gustavo Canuto Cunha

**INTEGRAÇÃO DE UM SISTEMA DE RECONHECIMENTO FACIAL
NA ASSISTÊNCIA A PESSOAS COM ALZHEIMER**

Ibirité - MG
2025

GUSTAVO CANUTO CUNHA

**INTEGRAÇÃO DE UM SISTEMA DE RECONHECIMENTO FACIAL
NA ASSISTÊNCIA A PESSOAS COM ALZHEIMER**

Trabalho de conclusão de curso apresentado ao Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - *Campus* Ibirité para a obtenção do título de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Diego Henrique de Souza Chaves

Ibirité - MG
2025

C972i
2025

Cunha, Gustavo Canuto.

Integração de um sistema de reconhecimento facial na assistência a pessoas com Alzheimer. [manuscrito] / Gustavo Canuto Cunha. – Ibirité: Instituto Federal de Minas Gerais, 2025.

79 f.; il.

Trabalho de Conclusão de Curso (Bacharelado em Engenharia de controle e automação) – Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – Campus Ibirité, 2025.

Orientador: Prof. Dr. Diego Henrique de Souza Chaves.

1. Automação. 2. Reconhecimento facial (Computação). 3. Alzheimer, Doença de. I. Chaves, Diego Henrique de Souza. II. Instituto Federal de Minas Gerais – *Campus* Ibirité. III. Título

CDU 004.93:616.831


Catálogo: Luciana Batista Neves - CRB-6/2000

Gustavo Canuto Cunha


INTEGRAÇÃO DE UM SISTEMA DE RECONHECIMENTO FACIAL NA ASSISTÊNCIA A PESSOAS COM ALZHEIMER

Trabalho de conclusão de curso apresentado ao Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - *Campus Ibirité* para a obtenção do título de Engenheiro de Controle e Automação.


Aprovado em: 27/02/2025 pela banca examinadora:

Documento assinado digitalmente
 DIEGO HENRIQUE DE SOUZA CHAVES
Data: 03/04/2025 15:38:35-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Diego Henrique de Souza Chaves - IFMG (Orientador)

Documento assinado digitalmente
 THIAGO HENRIQUE BARBOSA DE CARVALHO TA
Data: 02/04/2025 23:15:22-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Thiago Henrique Barbosa de Carvalho Tavares - IFMG

Documento assinado digitalmente
 CARLOS DIAS DA SILVA JUNIOR
Data: 03/04/2025 12:48:37-0300
Verifique em <https://validar.iti.gov.br>

Prof. Me. Carlos Dias da Silva Júnior - IFMG

Dedico esta monografia aos meus amados pais e irmãos,
maiores incentivadores e fontes inesgotáveis de apoio,
amor e compreensão.

AGRADECIMENTOS

Agradeço imensamente ao meu orientador, Diego Henrique de Souza Chaves, por quem nutro profundo respeito e admiração. Seu suporte, paciência e dedicação foram fundamentais ao longo dessa jornada acadêmica. Desde o início, sempre estive disposto a compartilhar seus conhecimentos, esclarecer dúvidas e incentivar meu crescimento, não apenas como estudante, mas também como profissional. Sua orientação cuidadosa e conselhos valiosos foram essenciais para a construção deste trabalho, e sou extremamente grato por toda a confiança e apoio que me proporcionou.

Minha gratidão também se estende aos meus pais, Aurila e Luiz, cuja presença, amor e dedicação foram pilares essenciais em minha vida. Desde cedo, me ensinaram a importância da educação, do esforço e da perseverança. Cada palavra de incentivo, cada gesto de apoio e cada sacrifício feito para que eu pudesse seguir meus estudos foram determinantes para que eu chegasse até aqui. Sem vocês, nada disso teria sido possível. Espero que este trabalho seja uma pequena forma de retribuir tudo o que sempre fizeram por mim.

Aos meus irmãos, Renan e Lincoln, agradeço imensamente pelo companheirismo, pela amizade e pelo incentivo constante. Compartilhamos desafios, aprendizados e momentos de superação, e saber que posso contar com vocês torna qualquer caminhada mais leve e significativa. Seus exemplos de determinação e suas palavras de motivação sempre foram fontes de inspiração para mim, e sou profundamente grato por tê-los ao meu lado.

Um agradecimento especial ao meu amigo Lorrán, cuja ajuda e suporte foram essenciais durante o desenvolvimento deste projeto. Sua disposição em compartilhar conhecimentos, esclarecer dúvidas e me apoiar nos momentos desafiadores fizeram toda a diferença nessa jornada. Ter alguém com quem contar nos momentos de dificuldade foi de imenso valor, e sou muito grato por sua amizade e parceria ao longo desse processo.

A todos que, direta ou indiretamente, contribuíram para minha formação acadêmica e pessoal, meus sinceros agradecimentos. Professores, amigos, colegas de curso e familiares, cada um, à sua maneira, teve um papel importante nessa trajetória. Sou grato por cada troca de conhecimento, por cada conversa enriquecedora e por todo apoio recebido ao longo desse percurso.

Por fim, agradeço a Deus pela força, saúde e pelas oportunidades que me permitiram chegar até aqui. Este trabalho é o reflexo de muito esforço e dedicação, mas também de todas as pessoas incríveis que me acompanharam nessa caminhada.

RESUMO

O presente trabalho teve como objetivo desenvolver um sistema de reconhecimento facial para auxiliar pessoas com Alzheimer no reconhecimento de familiares e cuidadores. A proposta busca oferecer uma ferramenta acessível que possa contribuir para a qualidade de vida dos pacientes, reduzindo a desorientação causada pela perda de memória. Para isso, foi criado um sistema capaz de identificar rostos em tempo real e anunciar verbalmente o nome da pessoa reconhecida, proporcionando um suporte auditivo que auxilia no reconhecimento. Além da identificação visual, foram realizados testes para avaliar o impacto de condições adversas, como imagens desfocadas, garantindo que o sistema pudesse funcionar mesmo em situações menos favoráveis. Também foram aplicadas medidas para evitar repetições excessivas das identificações, tornando o uso mais intuitivo e agradável para o usuário. Os dados obtidos demonstraram que o sistema é uma alternativa viável para auxiliar pessoas com Alzheimer, permitindo que reconheçam aqueles ao seu redor de maneira mais fácil e independente. A tecnologia utilizada mostrou-se eficiente e acessível, podendo ser aplicada em dispositivos simples, o que facilita sua adoção por famílias e instituições de cuidado.

Palavras-chave: Automação. Reconhecimento Facial. Alzheimer.

ABSTRACT

This work aimed to develop a facial recognition system to assist people with Alzheimer's in recognizing family members and caregivers. The proposed solution seeks to provide an accessible tool that can contribute to the quality of life of patients by reducing disorientation caused by memory loss. To achieve this, a system was created to identify faces in real time and verbally announce the name of the recognized person, offering auditory support to facilitate recognition. In addition to visual identification, tests were conducted to evaluate the impact of adverse conditions, such as blurred images, ensuring that the system could function even in less favorable situations. Measures were also implemented to prevent excessive repetition of identifications, making the experience more intuitive and user-friendly. The results demonstrated that the system is a viable alternative to assist people with Alzheimer's, enabling them to recognize those around them more easily and independently. The technology used proved to be efficient and accessible, allowing implementation on simple devices, which facilitates its adoption by families and care institutions.

Keywords: Automation. Facial Recognition. Alzheimer.

LISTA DE ILUSTRAÇÕES

| | |
|---|----|
| Figura 1 – Representação do sistema Whadda. | 19 |
| Figura 2 – Sistema de som com Arduíno. | 20 |
| Figura 3 – Óculos para dar assistência a pessoas com Alzheimer em estado leve | 22 |
| Figura 4 – Aplicativo de assistência a pessoas com Alzheimer | 23 |
| Figura 5 – ESP32-cam. | 24 |
| Figura 6 – Pinos do ESP32-cam. | 25 |
| Figura 7 – Sistema Hash. | 28 |
| Figura 8 – Lista de referências. | 32 |
| Figura 9 – Codificações associadas a cada nome. | 33 |
| Figura 10 – Funcionamento das bibliotecas. | 35 |
| Figura 11 – Esquemático do sistema. | 37 |
| Figura 12 – Teste de parentesco. | 39 |
| Figura 13 – Rosto 45°. | 42 |
| Figura 14 – Rosto 90°. | 43 |
| Figura 15 – Rosto 0°. | 43 |
| Figura 16 – Teste com uma fonte de luz. | 45 |
| Figura 17 – Teste com duas fontes de luz. | 45 |
| Figura 18 – Teste de blur. | 46 |
| Figura 19 – Desfoque de 0 a 60 (%). | 47 |
| Figura 20 – Desfoque de 80 a 99 (%). | 48 |
| Figura 21 – Exemplos da Tabela de Dados. | 49 |
| Figura 22 – Matriz de correlação. | 50 |
| Figura 23 – Gráfico Blur x Tempo de Reconhecimento. | 51 |
| Figura 24 – Tabela Blur x Acurácia. | 52 |
| Figura 25 – Gráfico Inclinação x Tempo de Reconhecimento. | 53 |
| Figura 26 – Gráfico Acurácia x Inclinação. | 54 |
| Figura 27 – Gráfico Graus de Tolerância X Tempo de Reconhecimento. | 54 |
| Figura 28 – Gráfico Acurácia x Graus de Tolerância. | 55 |
| Figura 29 – Gráfico de Pareto do Tempo de Reconhecimento. | 56 |
| Figura 30 – Gráfico de Interação: Blur x Tolerância x Inclinação no tempo de reconheci- mento. | 57 |
| Figura 31 – Gráfico de Interação: Tolerância x Inclinação no tempo de reconhecimento. | 57 |
| Figura 32 – Gráfico de Interação: Tolerância x Blur no tempo de reconhecimento. | 58 |
| Figura 33 – Gráfico de Interação: Blur x Inclinação no tempo de reconhecimento. | 58 |
| Figura 34 – Gráfico de Pareto da Acurácia. | 59 |
| Figura 35 – Gráfico de Interação: Tolerância x Inclinação na acurácia. | 60 |
| Figura 36 – Gráfico de Interação: Tolerância x Inclinação na acurácia. | 60 |
| Figura 37 – Gráfico de Interação: Tolerância x Blur na acurácia. | 61 |

Figura 38 – Gráfico de Interação: Blur x Inclinação na acurácia. 62

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 – Tabela de Custos dos Produtos (Cotação realizada em 07/01/2025) | 62 |
|--|----|

LISTA DE ABREVIATURAS E SIGLAS

| | |
|--------|-------------------------------------|
| IFMG | Instituto Federal de Minas Gerais |
| ML | Machine Learning |
| IA | Inteligência Artificial |
| IoT | Internet of Things |
| PSG | Polissonografia |
| OpenCV | Open Source Computer Vision Library |
| CNN | Convolutional Neural Networks |
| GPIO | General Purpose Input/Output |
| Pytsx3 | Python Text-to-Speech |
| JSON | JavaScript Object Notation |

SUMÁRIO

| | | |
|--------------|--|-----------|
| 1 | INTRODUÇÃO | 14 |
| 1.1 | Objetivos | 16 |
| <i>1.1.1</i> | <i>Objetivo geral</i> | 16 |
| <i>1.1.2</i> | <i>Objetivos específicos</i> | 16 |
| 1.2 | Justificativa | 16 |
| 1.3 | Organização do Texto | 16 |
| 2 | REVISÃO BIBLIOGRÁFICA | 18 |
| 3 | METODOLOGIA | 24 |
| 3.1 | ESP32-cam | 24 |
| 3.2 | Código | 26 |
| <i>3.2.1</i> | <i>MediaPipe</i> | 26 |
| <i>3.2.2</i> | <i>Sistema Hash</i> | 27 |
| <i>3.2.3</i> | <i>Open CV</i> | 29 |
| <i>3.2.4</i> | <i>Pytsx3</i> | 30 |
| <i>3.2.5</i> | <i>JSON</i> | 31 |
| <i>3.2.6</i> | <i>Aplicação no Código</i> | 33 |
| 3.3 | Montagem do produto | 36 |
| 3.4 | Experimento e Análise dos dados | 37 |
| 4 | RESULTADOS | 39 |
| 4.1 | Reconhecimento de pessoas distintas | 39 |
| 4.2 | Sistema de Voz | 40 |
| 4.3 | Níveis de tolerância | 40 |
| 4.4 | Angulação | 42 |
| 4.5 | Teste de luminosidade | 44 |
| 4.6 | Teste de blur | 46 |
| 4.7 | Análise dos Dados | 48 |
| 4.8 | Análise Estatística | 56 |
| 4.9 | Custos e viabilidade | 62 |
| 5 | CONCLUSÃO | 64 |

| | | |
|-------------|---|-----------|
| 5.1 | Trabalhos Futuros | 65 |
| | REFERÊNCIAS | 67 |
| .1 | Códigos | 69 |
| .1.1 | <i>Código de Cadastro</i> | 69 |
| .1.2 | <i>Código de Reconhecimento Facial</i> | 70 |
| .1.3 | <i>Código da Câmera</i> | 73 |

1 INTRODUÇÃO

Como parte do desenvolvimento urbano, a busca por automação nas diversas áreas da sociedade passou a se normalizar, otimizando o tempo gasto em tarefas rotineiras e repetitivas. Nesse viés, como constatado por (Beller *et al.*, 2018), a utilização de controle, automação, robótica e técnicas de medição têm sido fundamentais para o desenvolvimento da indústria nas últimas décadas, possuindo um potencial inovador e significativo, não sendo o campo da saúde uma exceção. Assim, ferramentas únicas e versáteis têm surgido cada vez mais, desde aplicativos de auxílio, até equipamentos médicos automatizados, o que é extremamente crucial, principalmente se tratando do combate aos sintomas do Alzheimer.

O Alzheimer é uma doença neurodegenerativa progressiva e irreversível que afeta principalmente a memória, o pensamento e o comportamento, sendo a forma mais comum de demência em todo o mundo. (Scheltens *et al.*, 2016) destaca que essa doença caracteriza-se pelo acúmulo anormal de placas beta-amiloides e emaranhados neurofibrilares no cérebro, que resultam em degeneração neuronal e comprometimento das funções cognitivas. Os sintomas iniciais incluem lapsos de memória e dificuldade em realizar tarefas cotidianas, progredindo para perda severa de habilidades cognitivas e autonomia funcional.

(Sereniki; Vital, 2008) ressaltam o impacto dessa doença na sociedade atual ao dizer que ela afeta aproximadamente 10% dos indivíduos com idade superior a 65 anos e 40% acima de 80 anos, estimando que, em 2050, mais de 25% da população mundial será idosa, o que contribui para uma maior prevalência do Alzheimer. Embora não haja cura definitiva, intervenções terapêuticas e tecnológicas têm sido visadas para retardar o avanço da doença e melhorar a qualidade de vida dos pacientes e seus cuidadores, sendo a automação uma área de suma importância para atingir esse propósito.

(Elif; Taşkapılıoğlu; Bakar, 2017) constata que essa doença pode ser dividida em três estágios: inicial, intermediário e avançado. No estágio inicial, os sintomas são sutis e muitas vezes negligenciados, incluindo lapsos de memória, perda da noção do tempo e dificuldade em aprender novas informações, embora a independência do paciente ainda seja preservada. No estágio intermediário, os sintomas tornam-se mais evidentes e debilitantes, com dificuldades em tarefas básicas, desorientação, alterações comportamentais e transtornos psiquiátricos, como paranoia e agressividade, aumentando significativamente o fardo sobre os cuidadores. O estágio avançado é caracterizado pela total dependência do paciente, incapacidade de realizar atividades básicas, perda de reconhecimento de familiares e complicações médicas graves, sendo a fase mais desgastante para os cuidadores, tanto emocional quanto fisicamente.

A integração da automação com a medicina assistiva permite uma maior independência e conforto na rotina de pacientes com Alzheimer em estágio inicial, como demonstrado por (Aljojo *et al.*, 2020), que utilizou do monitoramento em tempo real, propiciado pela integração de hardwares cotidianos com o GPS, para permitir maior contato entre cuidador e paciente, entre outras soluções mais tecnológicas, como a integração total de ambientes com sensores a fim de

relembrar pessoas com essa doença sobre a funcionalidade de cada cômodo e o propósito dele no respectivo local.

A utilização de tecnologias emergentes como aprendizado de máquina, reconhecimento facial e dispositivos IoT (Internet das Coisas) tem sido um dos focos principais no desenvolvimento de ferramentas para pacientes com Alzheimer. Esses avanços permitem não apenas a identificação precoce da doença, mas também oferecem soluções para mitigar os impactos de seus sintomas. Sistemas baseados em inteligência artificial, por exemplo, têm sido empregados para analisar padrões comportamentais e prever episódios de desorientação, auxiliando cuidadores e familiares na tomada de decisões mais ágeis e informadas.

Além disso, dispositivos portáteis, como smartwatches e óculos equipados com câmeras e sistemas de processamento, podem ser projetados para auxiliar pacientes a reconhecer rostos familiares e acessar informações contextuais. Essas tecnologias promovem uma interação mais humanizada e adaptada às necessidades dos indivíduos, reduzindo a ansiedade gerada pela confusão e fortalecendo os laços sociais, que muitas vezes são prejudicados pela progressão da doença.

O uso de tecnologias assistivas não se limita ao apoio direto ao paciente. Plataformas de monitoramento remoto, integradas a aplicativos móveis, permitem que cuidadores acompanhem o estado de saúde e as atividades dos pacientes em tempo real. Essas soluções incluem alertas para situações de risco, como quedas ou deslocamentos inesperados, aumentando a segurança e a tranquilidade para ambos os lados.

Paralelamente, dada a busca pela democratização do acesso às ferramentas tecnológicas para pacientes e cuidadores, soluções de baixo custo, como dispositivos baseados no microcontrolador ESP32, se tornaram mais frequentes, possibilitando a implementação de sistemas eficientes e acessíveis, que atendem tanto às demandas individuais quanto às limitações financeiras de diferentes perfis de usuários. Isso contribui para a inclusão de uma parcela maior da população em uma era de automação e tecnologia assistiva.

Diante desse cenário, o contínuo avanço das tecnologias de automação e sua aplicação no campo médico destacam o potencial transformador dessas ferramentas. Com o envelhecimento populacional em escala global, é fulcral que soluções inovadoras sejam desenvolvidas e aplicadas, garantindo qualidade de vida e autonomia aos pacientes, além de proporcionar suporte aos cuidadores. O uso integrado de automação, inteligência artificial e dispositivos conectados se apresenta como uma abordagem para enfrentar os desafios impostos pelo Alzheimer e outras condições relacionadas ao envelhecimento.

1.1 Objetivos

1.1.1 *Objetivo geral*

O objetivo geral do trabalho é desenvolver um sistema assistivo de reconhecimento facial, baseado no microcontrolador ESP32, para auxiliar pacientes com Alzheimer em estágio inicial no reconhecimento de rostos familiares e no acesso a informações contextuais, promovendo maior independência, segurança e qualidade de vida.

1.1.2 *Objetivos específicos*

1. Projetar e implementar um sistema de reconhecimento facial integrado a dispositivos IoT, utilizando o ESP32 como base para garantir um baixo custo e acessibilidade.
2. Testar a viabilidade do sistema em cenários simulados ou reais, avaliando sua eficácia e desempenho.
3. Propor soluções escaláveis e acessíveis, que possam ser implementadas de maneira prática e replicável, contribuindo para a democratização das tecnologias assistivas.

1.2 Justificativa

O Alzheimer é uma das principais causas de perda de autonomia em idosos, afetando tanto os pacientes quanto seus cuidadores e familiares. Com a crescente prevalência da doença devido ao envelhecimento populacional, a demanda por soluções inovadoras e acessíveis tem se tornado urgente. Tecnologias assistivas, como sistemas de reconhecimento facial integrados a dispositivos IoT, oferecem uma oportunidade única para aliviar os impactos da doença.

Além de promover maior independência aos pacientes, essas tecnologias auxiliam cuidadores ao fornecer informações em tempo real sobre deslocamentos e comportamentos, garantindo maior segurança e tranquilidade. O uso de um microcontrolador como o ESP32 permite a criação de dispositivos eficientes e de baixo custo, acessíveis a uma parcela maior da população.

Este trabalho se justifica pela relevância social e tecnológica de contribuir para a melhoria da qualidade de vida de pessoas com Alzheimer, além de fomentar o avanço das tecnologias assistivas em uma sociedade cada vez mais envelhecida e dependente de soluções acessíveis.

1.3 Organização do Texto

O trabalho está estruturado da seguinte forma:

- **Capítulo 1 – Introdução:** Apresenta o contexto geral da pesquisa, a importância do tema e os objetivos do trabalho.

- **Capítulo 2 – Revisão Bibliográfica:** Aborda conceitos relacionados ao Alzheimer, tecnologias assistivas, reconhecimento facial, IoT e o microcontrolador ESP32, destacando os avanços e desafios na área.
- **Capítulo 3 – Metodologia:** Detalha os métodos utilizados no desenvolvimento do sistema, incluindo escolha de ferramentas, técnicas de reconhecimento facial, integração de hardware e software e critérios para avaliação do desempenho.
- **Capítulo 4 – Resultados e Discussão:** Apresenta os resultados obtidos com o sistema proposto, análise de desempenho, viabilidade e impactos na rotina de pacientes e cuidadores.
- **Capítulo 5 – Conclusões e Trabalhos Futuros:** Resgata os objetivos, destaca as contribuições do trabalho e propõe possíveis melhorias e avanços futuros na área de tecnologias assistivas.

2 REVISÃO BIBLIOGRÁFICA

No contexto do desenvolvimento da automação, a implementação de ambientes que fazem uso de tecnologias IoT tem se tornado cada vez mais comum para diversas áreas da sociedade, tendo extrema importância na atualidade. (Wang *et al.*, 2020) descreve o IoT como sendo a interconexão e interoperabilidade entre dispositivos cotidianos, bem como a autonomia do dispositivo, percepção e consciência situacional, ressaltando a eficiência com que tais dispositivos atuam no meio.

Diante desse cenário, a interconexão entre os campos da automação e a medicina tem se tornado cada vez mais comum, propiciando melhorias contínuas em tarefas repetitivas, bem como auxiliando os profissionais. (Rezaei *et al.*, 2023) retratam isso em seu estudo, reforçando que a área da saúde está passando por uma transformação com o uso de tecnologias como inteligência artificial (IA), robótica e automação, seja para analisar grandes volumes de dados médicos, auxiliar em diagnósticos precisos, tratamentos personalizados ou até mesmo para previsões de resultados.

(Alattar; Govind; Mainali, 2024) exemplificam essa integração entre automação e a área médica, afirmando que a medicina do sono enfrenta desafios significativos com métodos tradicionais de diagnóstico, como a polissonografia (PSG), que são caros e trabalhosos. Tendo isso em vista, os autores ressaltam que inteligência artificial (IA) surgiu como uma solução promissora, capaz de processar grandes volumes de dados clínicos, identificar padrões complexos e automatizar a análise de exames do sono, servindo de auxílio para profissionais desse campo, embora ainda existam algumas barreiras de implementação e acesso a dados diversificados.

(Raparathi, 2020) explora como a combinação entre a automação de processos e robóticos e uso de inteligências artificiais estão aprimorando os fluxos de trabalho da medicina de precisão, tanto na área de atendimento quanto nos resultados dos tratamentos. O autor reforça que a automatização de tarefas repetitivas, é capaz de reduzir erros humanos e aumentar a eficiência operacional, enquanto a IA aprimora a tomada de decisões, analisando dados complexos e fornecendo auxílio. Dessa maneira, as duas tecnologias, ao trabalharem em sincronia, podem melhorar os processos na medicina, de maneira a permitir que os profissionais de saúde ofereçam um atendimento mais personalizado e oportuno.

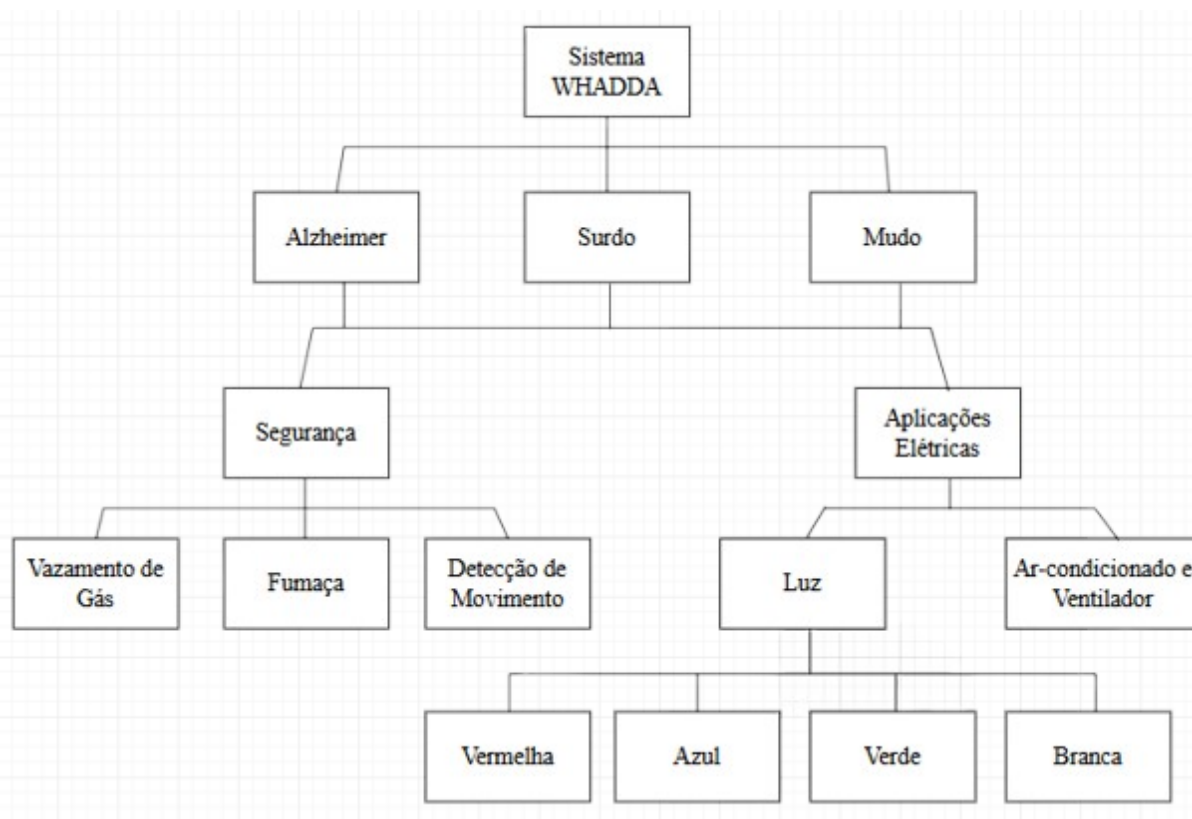
(Naik; Bhide, 2014) reforçam isso no avanço da tecnologia de sequenciamento genômico, salientando que, dada a quantidade massiva de dados que exigem novas abordagens, a automação do trabalho de conhecimento, impulsionada por inteligência artificial, desponta como solução para organizar, interpretar e aplicar esses dados de forma eficiente.

Diante do exposto, uma das camadas da medicina que tem sido beneficiada pela incorporação de sistemas de automação diz respeito ao tratamento de pessoas com Alzheimer, uma doença neurodegenerativa progressiva caracterizada pela deterioração cognitiva, perda de memória e mudanças comportamentais, causada pelo acúmulo anormal de proteínas no cérebro,

como beta-amiloide e tau. Apesar de ainda não possuir cura, as ferramentas adquiridas por meio da associação entre a área da saúde e sistemas de automação têm se mostrado extremamente úteis, tanto para proporcionar conforto a pacientes que apresentam esses sintomas, quanto para ajudá-los em tarefas rotineiras e reconhecer entes familiares. (Obe; Abe; Boyinbode, 2018) demonstram isso em seu estudo, abordando o crescimento significativo da população com deficiências, atualmente estimada em 15% da população global, e destacando a necessidade de tecnologias que promovam autonomia e segurança. Dessa maneira, a incorporação de um sistema voltado para pessoas com deficiências como surdez, mudez e Alzheimer, nomeado de WHADDA, utilizou da automação residencial para monitorar o ambiente, controlar dispositivos elétricos e alertar sobre perigos, como vazamentos de gás ou intrusões, se mostrando eficiente no auxílio desse grupo.

O sistema WHADDA integrou sensores, atuadores e uma interface baseada na web para gerenciamento via dispositivos móveis, tal qual tecnologias como sensores de gás, fumaça e movimento, junto a alertas visuais e sonoros, aumentaram a segurança e a praticidade para os usuários, ilustrando a viabilidade da automação na melhoria da qualidade de vida de grupos vulneráveis. Portadores de necessidades especiais também poderiam gerenciar os aparelhos elétricos em suas casas, como luzes, ar-condicionado e ventiladores por meio de dispositivos móveis ou tablets com acesso à internet, como demonstrada pela Figura 1.

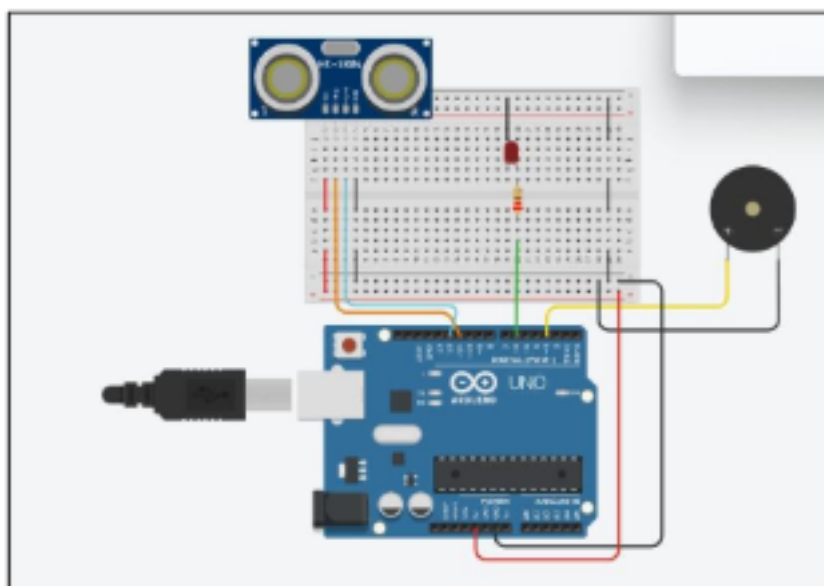
Figura 1 – Representação do sistema Whadda.



Adaptado de: (Obe; Abe; Boyinbode, 2018);

(Bernardo; Silva; Lopes, 2023) também apresentam uma proposta inovadora para auxiliar pessoas com Alzheimer, construindo um protótipo com a plataforma Arduino que possui o intuito de fornecer apoio com a identificação de cômodos na residência, demonstrado na Figura 2. O dispositivo, ao ser instalado na parte superior do batente da porta de cada cômodo da residência, é capaz de emitir um sinal de voz, que informará qual é a finalidade daquele ambiente e quais atividades costumam ser realizadas nele, se mostrando uma solução interessante para prestar ajuda com os sintomas da doença.

Figura 2 – Sistema de som com Arduino.



Fonte: (Bernardo; Silva; Lopes, 2023)

A integração da automação na medicina também é capaz de auxiliar no processo de identificação dessa doença, tal como demonstrado por (Joshi *et al.*, 2025) em seu estudo. Os autores enfatizam o uso do eletroencefalograma (EEG) como uma ferramenta valiosa, rápida, acessível e econômica para detectar demência em estágios iniciais, sendo a identificação precoce crucial para a realização de intervenções mais eficazes e com menos riscos. Assim, ao utilizar o aprendizado de máquina (Machine Learning) em consonância com os dados do electroencefalograma, foi possível realizar um classificador com excelente desempenho, alcançando 97,27% de precisão, o que contribui em muito com o tratamento.

(Lodha; Talele; Degaonkar, 2018) também demonstram essa integração entre automação e identificação do Alzheimer em seu estudo, reforçando o uso de técnicas como exames de neuroimagens para a identificação da doença. De maneira similar, também é proposto o uso de algoritmos de aprendizado de máquina para processar dados obtidos por essas tecnologias, visando detectar o Alzheimer em estágios iniciais.

Considerando as diversas possibilidades do uso da automação, o reconhecimento facial tem se destacado como uma ferramenta extremamente versátil no campo da medicina, auxiliando

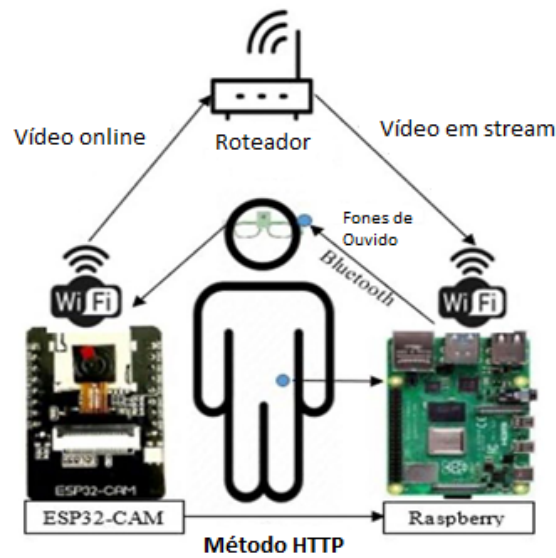
em diversas tarefas. (Chan *et al.*, 2024) exploram isso em seu trabalho, aprofundando-se no uso do processamento facial no campo da saúde, detalhando como o reconhecimento facial e tecnologias relacionadas são aplicados em diferentes especialidades médicas e ressaltando sua utilidade.

Em se tratando de pacientes com Alzheimer, a identificação de familiares também se torna um grande problema dado os sintomas dessa doença. Pensando nisso, (Roopaei; Rad; Prevost, 2018) propuseram um dispositivo vestível baseado em IoT que utiliza o aprendizado de máquina para auxiliar essas pessoas com problemas de percepção facial. Apesar das limitações computacionais de dispositivos anteriores, como tempo de processamento e armazenamento de dados, os experimentos demonstraram que o sistema desenvolvido é capaz de identificar rostos conhecidos quase em tempo real, o que incentiva estudos mais aprofundados nesse campo e demonstra as vantagens da utilização de dispositivos capazes de realizar reconhecimento facial.

De forma semelhante, (Fardoun; Mashat; Castillo, 2017) também exploraram a ideia de apresentar uma tecnologia assistiva prototípica para pacientes com Alzheimer, visando ajudá-los a lembrar detalhes pessoais de pessoas familiares em sua rotina. Ao combinar um sistema de informações pessoais com reconhecimento facial, associado a uma interação via um smartwatch, a tecnologia permitiu que pacientes em estágios moderados e avançados da doença reconhecessem familiares e fossem capazes de acessar informações sobre eles, facilitando suas vidas diárias, melhorando a autoestima e promovendo estímulo com o uso de novas tecnologias.

Em consonância, (Kadhim *et al.*, 2023) também visaram auxiliar pacientes com Alzheimer na identificação de familiares por meio da utilização de dispositivos IoT. Os autores propuseram o desenvolvimento de um sistema portátil composto por óculos equipados com ESP32-CAM e um microcomputador Raspberry Pi. O sistema tinha o intuito de ser operado automaticamente, sem interação direta do usuário, capturando quadros de vídeo em tempo real e transmitindo-os via Wi-Fi para o Raspberry Pi, o qual seria responsável por reconhecer os rostos e fornecer feedback por áudio Bluetooth com o nome da pessoa identificada, como ilustrado pela Figura 3.

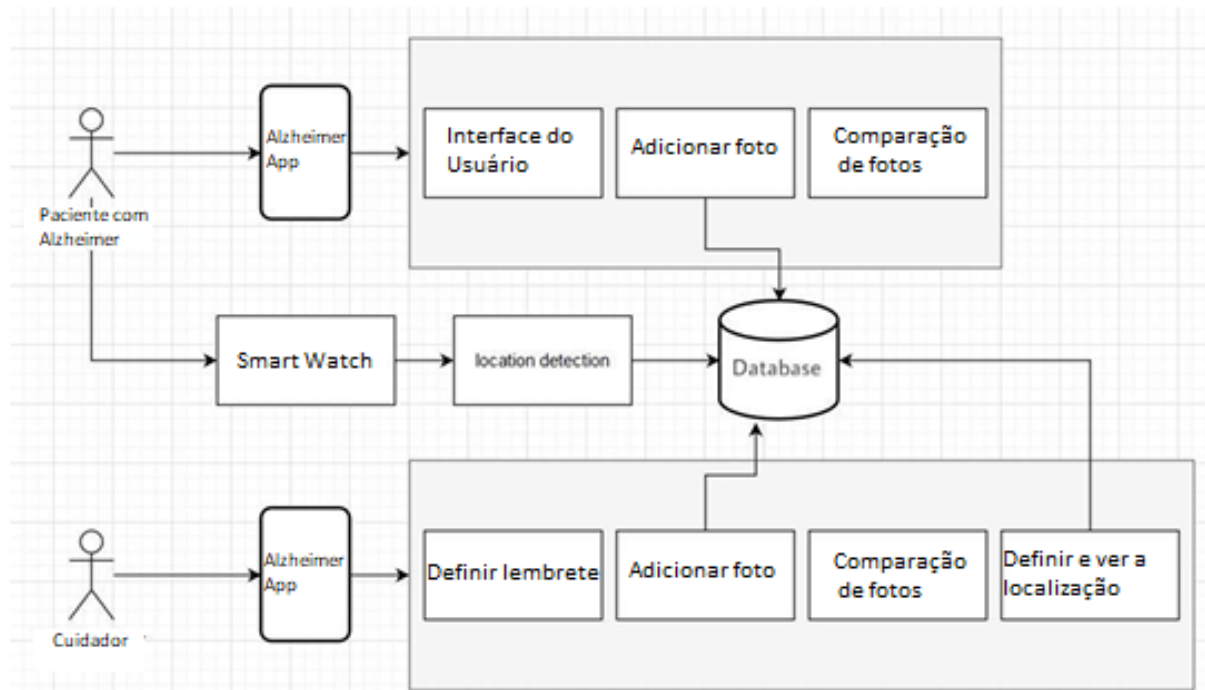
Figura 3 – Óculos para dar assistência a pessoas com Alzheimer em estado leve



Fonte: (Kadhim *et al.*, 2023); Tradução do autor

(Aljojo *et al.*, 2020) também apresentam uma proposta eficiente ao desenvolverem a criação de um aplicativo móvel capaz de auxiliar pacientes com Alzheimer em estágios iniciais e moderados a manterem sua independência e segurança. O aplicativo utiliza tecnologias de reconhecimento facial e detecção de localização via GPS para apoiar pacientes e seus cuidadores, sendo capaz de identificar pessoas com base em fotos e informações sobre o relacionamento do paciente com elas e enviar lembretes de tarefas diárias, além de também incorporar uma pulseira com rastreamento ao usuário a fim de evitar que pacientes se percam, se mostrando extremamente versátil, como demonstrado pela Figura 4.

Figura 4 – Aplicativo de assistência a pessoas com Alzheimer



Fonte: (Aljojo *et al.*, 2020); Tradução do autor

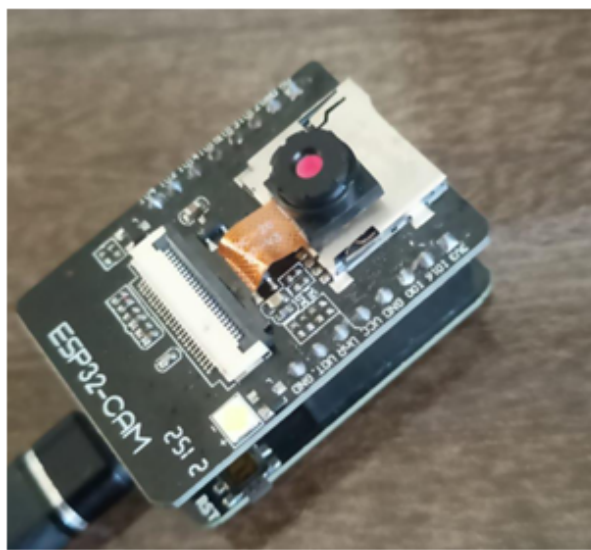
Diante dos avanços tecnológicos e da necessidade de soluções para melhorar a vida de pacientes com Alzheimer, o presente projeto busca desenvolver um sistema que utiliza reconhecimento facial como ferramenta central.

3 METODOLOGIA

3.1 ESP32-cam

O desenvolvimento do sistema de reconhecimento facial para assistência a pessoas com Alzheimer foi viabilizado principalmente pelo uso do ESP32-CAM, demonstrado na figura 6, um dispositivo compacto, econômico e amplamente funcional. A escolha desse microcontrolador foi baseada em suas capacidades técnicas e na viabilidade econômica para o desenvolvimento de projetos assistivos de baixo custo.

Figura 5 – ESP32-cam.



Fonte: Autoria própria.

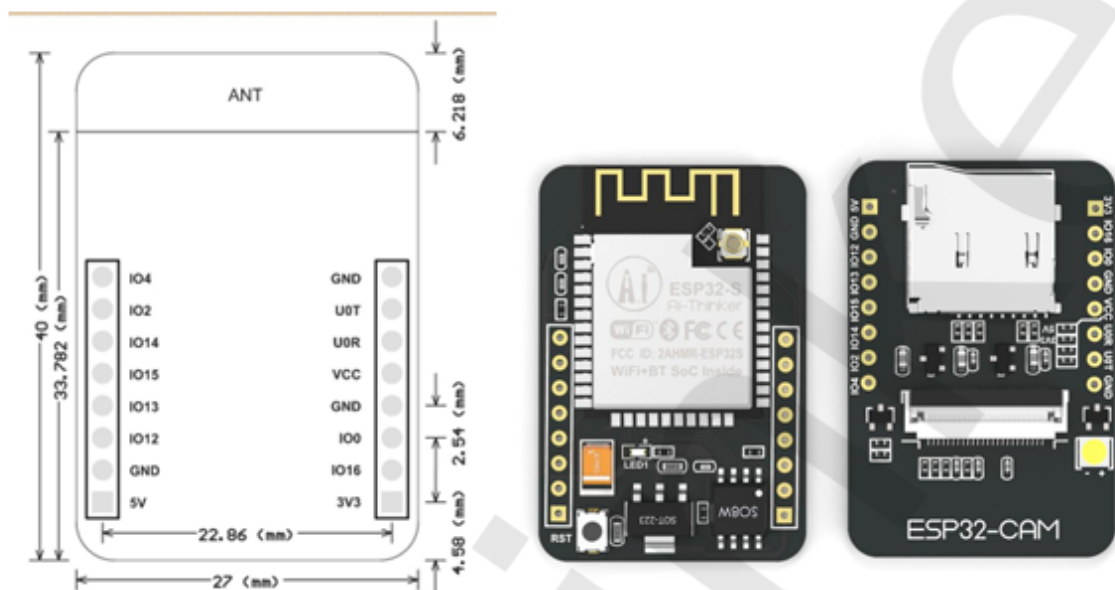
O ESP32-CAM é equipado com um microcontrolador ESP32, que integra uma unidade de processamento de alta performance, conectividade Wi-Fi e Bluetooth, além de recursos especializados para processamento de imagens. O data sheet do ESP32 destaca que o dispositivo possui dois núcleos de processamento (Xtensa® Dual-Core 32-bit LX6), operando a uma frequência de até 240 MHz, com uma memória RAM de 520 KB e suporte a armazenamento externo em cartões microSD de até 4 GB. Essa configuração permite que ele execute algoritmos de reconhecimento facial em tempo real com eficiência.

Adicionalmente, o ESP32-CAM inclui uma câmera OV2640 de 2 megapixels, que captura imagens com resolução ajustável de até 1600 x 1200 pixels. A câmera é responsável por coletar os dados visuais necessários para o reconhecimento facial, fornecendo imagens que são processadas diretamente no dispositivo. Essa funcionalidade é essencial para o sistema, pois permite identificar rostos armazenados no banco de dados e emitir alertas ou exibir informações contextuais.

Com dimensões compactas, aproximadamente 27 mm de largura por 40 mm de altura, o ESP32-CAM se provou versátil e capaz de integrar com dispositivos portáteis. Essa caracterís-

tica permitiu que o sistema fosse implementado de forma discreta, viabilizando sua aplicação no cotidiano de pacientes com Alzheimer. O microcontrolador é equipado com pinos GPIO (General Purpose Input/Output) que permitem a conexão de periféricos e sensores adicionais, bem como a integração com outros sistemas de automação. Além disso, conta com um slot para cartão microSD, que possibilita o armazenamento de imagens capturadas ou dados relevantes de forma local.

Figura 6 – Pinos do ESP32-cam.



ESPRESSIF SYSTEMS. ESP32-CAM: Camera module with Wi-Fi and Bluetooth. Disponível em: https://components101.com/sites/default/files/component_datasheet/ESP32-CAM_datasheet.pdf.

O processo de implementação do sistema foi dividido em etapas, começando pela configuração inicial do hardware. O ESP32-CAM foi integrado a um circuito que inclui módulos adicionais, como LEDs para iluminação, sensores de movimento para ativar a captura de imagens e um cartão microSD para armazenamento de dados. A alimentação do dispositivo foi realizada por meio de uma fonte de 5V, com capacidade para manter a estabilidade energética necessária para o processamento contínuo.

No que se refere ao software, o sistema foi desenvolvido utilizando a biblioteca `face_recognition`, a qual possui compatibilidade com o ESP32. Essa biblioteca foi utilizada para implementar reconhecimento facial e inclui módulos para captura, detecção e comparação de rostos. O primeiro passo foi configurar a câmera do ESP32-CAM para capturar imagens em tempo real. Em seguida, os algoritmos de detecção facial foram treinados para identificar características únicas dos rostos, como a distância entre os olhos, o formato do nariz e a curva do maxilar. Os dados extraídos foram armazenados em um banco de dados local, permitindo o reconhecimento em tempo real sem a necessidade de conexão constante à internet.

O sistema criado buscou se mostrar eficaz em reconhecer rostos armazenados no banco de dados, com uma taxa de acurácia satisfatória considerando o hardware utilizado. A capacidade de capturar imagens e processá-las localmente buscou garantir um desempenho rápido e estável, enquanto a conectividade Wi-Fi visou permitir a comunicação contínua com cuidadores.

O ESP32-CAM demonstrou ser uma escolha estratégica para o projeto, principalmente devido ao seu custo reduzido, funcionalidade integrada e facilidade de programação. Sua utilização no sistema de reconhecimento facial permitiu a criação de uma ferramenta assistiva acessível, que pode ser amplamente utilizada para melhorar a qualidade de vida de pessoas com Alzheimer e reduzir a carga de trabalho dos cuidadores. Além disso, a escalabilidade do dispositivo oferece possibilidades para futuros aprimoramentos, como a integração com serviços de nuvem e a expansão do banco de dados facial, garantindo que o sistema possa evoluir conforme as necessidades dos usuários.

3.2 Código

3.2.1 MediaPipe

A biblioteca MediaPipe, desenvolvida pelo Google, é uma ferramenta de código aberto amplamente utilizada em aplicações de visão computacional e aprendizado de máquina. (Lugaresi *et al.*, 2019) ressalta que sua principal função é fornecer soluções para a detecção e rastreamento de padrões em imagens e vídeos, com destaque para o reconhecimento de gestos, identificação facial, detecção de mãos, corpos e até mesmo objetos específicos. Essa biblioteca combina algoritmos com modelos de aprendizado de máquina pré-treinados, permitindo o desenvolvimento de aplicações robustas e de baixo custo computacional.

No contexto do desenvolvimento de um sistema de reconhecimento facial voltado para a assistência a pessoas com Alzheimer, o MediaPipe apresenta várias vantagens que o tornam uma escolha atrativa, sendo uma das suas principais funcionalidades o Face Mesh, uma solução capaz de detectar e mapear em tempo real mais de 460 pontos na face humana. Isso possibilita a criação de um modelo tridimensional detalhado do rosto, que pode ser usado para identificar com precisão características únicas de cada indivíduo.

Dessa maneira, o MediaPipe foi utilizado para melhorar a eficiência na tarefa de distinguir objetos de pessoas e na medição da diferença entre rostos comparados, até mesmo em casos de semelhança parental. Isso é essencial para permitir que o sistema discirna indivíduos, mesmo em casos de semelhanças físicas devido a parentesco.

Além disso, como MediaPipe é aplicado para dispositivos de baixo custo, como o ESP32, que pode ser utilizado em sistemas de Internet das Coisas (IoT), sua capacidade de realizar detecções em tempo real, mesmo em hardwares limitados, tornou a biblioteca ideal para sistemas que precisam ser implementados em residências ou dispositivos portáteis sem depender de servi-

dores externos ou recursos computacionais avançados. Essa abordagem foi crucial para garantir que o sistema fosse mais acessível e de fácil manutenção, um fator essencial no desenvolvimento de tecnologias assistivas.

Outro ponto importante é que o MediaPipe pode ser combinado com outras tecnologias, como o reconhecimento de voz ou a criação de aplicativos móveis. Isso permite a integração de múltiplas interfaces de interação, possibilitando, por exemplo, que o sistema forneça lembretes ou alertas quando um rosto desconhecido for detectado, ou mesmo para informar o paciente sobre o contexto de sua rotina diária.

Por fim, a utilização do MediaPipe também pode proporcionar escalabilidade e flexibilidade ao sistema. A biblioteca é compatível com diversas linguagens de programação, como Python e C++, e pode ser integrada a frameworks populares de aprendizado de máquina, como TensorFlow e PyTorch. Isso significa que o sistema pode ser facilmente expandido ou adaptado para incluir novas funcionalidades no futuro, como a análise de expressões faciais para identificar estados emocionais.

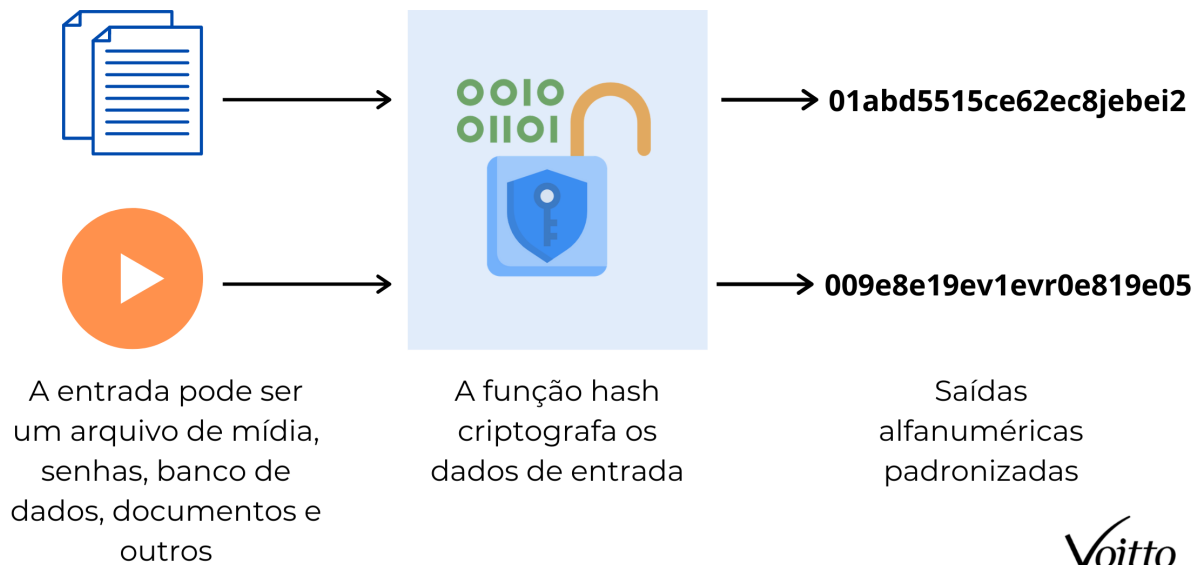
3.2.2 Sistema Hash

Com o uso da biblioteca MediaPipe, o sistema consegue realizar o primeiro protocolo do projeto desejado, que foi o de identificar e analisar faces. No entanto, para tornar o projeto funcional, é necessário implementar um banco de dados capaz de armazenar informações sobre as pessoas analisadas, o que seria fundamental para fornecer informações sobre os familiares aos portadores de Alzheimer.

Buscando lidar com o desafio de armazenar informações de forma eficiente e garantir que o servidor não ficasse sobrecarregado devido à quantidade de imagens que precisavam ser armazenadas, o sistema de Hash é uma alternativa eficaz.

O sistema de hash é uma técnica amplamente utilizada na ciência da computação para transformar grandes quantidades de dados em um formato compacto, chamado de valor de hash ou hash code. (Liu; Kadir; Liu, 2019) descrevem o sistema de hash como uma função matemática que transforma uma entrada de dados em uma representação única de tamanho fixo, chamada valor de hash, usada para garantir a integridade e a autenticidade dos dados, tal como demonstrado pela figura 7.

Figura 7 – Sistema Hash.



Fonte: VOITTO. O que é hash e como funciona?. Disponível em: <https://voitto.com.br/blog/artigo/o-que-e-hash-e-como-funciona>.

Essa transformação é realizada por meio de funções matemáticas conhecidas como funções hash, que recebem uma entrada, como um texto ou uma imagem, e produzem uma sequência de caracteres de tamanho fixo. Um dos principais objetivos de um sistema de hash é garantir que mesmo pequenas alterações na entrada resultem em valores de hash completamente diferentes, o que confere segurança e eficiência ao método.

Os valores de hash ocupam significativamente menos espaço de armazenamento do que as informações completas de cada rosto, já que se tratam de valores hexadecimais. Em um sistema projetado para rodar em dispositivos de baixo custo, como o ESP32 ou servidores com recursos limitados, a compactação oferecida pelos valores de hash é essencial para garantir eficiência no uso de memória e armazenamento. Isso permite que o sistema armazene dados sobre um maior número de rostos e informações associadas, como o nome da pessoa, parentesco, histórico de visitas ou mensagens personalizadas, sem sobrecarregar os recursos disponíveis.

Além disso, o sistema de hash contribui para a segurança dos dados pessoais. Em um ambiente de assistência para pessoas com Alzheimer, onde informações pessoais e médicas são frequentemente utilizadas, garantir a privacidade dos usuários é fundamental. Como o valor de hash é uma representação irreversível, ele pode ser armazenado no banco de dados sem risco de que o rosto original ou outras informações detalhadas sejam facilmente reconstruídos a partir dele. Isso significa que, mesmo que o banco de dados seja acessado de maneira não autorizada, os dados originais permanecem protegidos.

Outra aplicação prática do sistema de hash no reconhecimento facial é a aceleração do processo de identificação. Em vez de comparar diretamente a imagem de entrada com todas as imagens armazenadas, o sistema pode calcular o hash da nova imagem e compará-lo rapidamente com os valores de hash existentes. Essa operação reduz significativamente o tempo de processamento, especialmente quando o banco de dados contém um grande número de registros.

Assim, os sistemas de hash se destacam como uma ferramenta poderosa no desenvolvimento de tecnologias voltadas para o reconhecimento facial em contextos assistivos. Eles permitem o armazenamento otimizado de informações, aumentam a segurança dos dados e melhoram o desempenho do sistema. No caso de pacientes com Alzheimer, esses benefícios se traduzem em soluções mais escaláveis e acessíveis, que podem ser implementadas em residências ou dispositivos portáteis, promovendo um cuidado mais eficiente e humanizado.

3.2.3 Open CV

A OpenCV (Open Source Computer Vision Library) é uma biblioteca de código aberto amplamente utilizada no campo da visão computacional e processamento de imagens. Essa biblioteca é uma ferramenta eficiente para implementar soluções que envolvam análise e interpretação de imagens e vídeos, possuindo interfaces para linguagens de programação, como Python, Java e MATLAB, o que a torna versátil e acessível para diferentes plataformas.

Tal como reforçado por (Bradski, 2000), a biblioteca oferece uma ampla gama de funcionalidades voltadas para o processamento de imagens, reconhecimento de padrões, análise de objetos e movimento, além de algoritmos de aprendizado de máquina aplicados à visão computacional. Dentre suas principais capacidades, destacam-se operações como leitura, manipulação e exibição de imagens, transformação geométrica, detecção de bordas e contornos, análise de formas e texturas, detecção de objetos, rastreamento de movimento e calibração de câmeras.

No contexto de reconhecimento facial, a OpenCV apresenta recursos específicos que tornam o desenvolvimento de sistemas desse tipo mais prático e eficiente. Ela oferece implementações otimizadas de algoritmos de detecção de rostos e modelos baseados em redes neurais convolucionais (CNNs) para tarefas mais avançadas. Além disso, a biblioteca possui ferramentas para detecção de pontos-chave no rosto, como a distância entre nariz e olhos, que são de extrema importância para análise de expressões faciais, reconhecimento de identidade e até mesmo rastreamento de emoções. Essas funcionalidades podem ser facilmente combinadas para atender a diferentes aplicações, como autenticação de usuários, segurança, personalização de experiências em dispositivos e automação residencial.

No desenvolvimento de um sistema de reconhecimento facial, a OpenCV não é responsável somente pela detecção e identificação de rostos, mas também auxilia no pré-processamento das imagens. Desse modo, ela possibilita a normalização de imagens, como ajuste de contraste e brilho, e alinhamento dos rostos detectados para garantir maior consistência no reconhecimento.

Uma das grandes vantagens da OpenCV é sua integração com outras ferramentas e bibliotecas, como o MediaPipe, que foi utilizado no projeto em questão. Essa integração permitiu combinar o poder de detecção rápida e precisa de rostos do MediaPipe com os recursos avançados de processamento de imagens da OpenCV. O resultado foi um sistema melhorado, capaz de realizar a identificação facial com maior velocidade e menor consumo de recursos computacionais. Por exemplo, enquanto o MediaPipe identifica as regiões faciais, a OpenCV pode realizar análises mais detalhadas, como reconhecimento de características específicas.

Outro ponto de destaque da OpenCV é sua capacidade de operar em diferentes plataformas, incluindo dispositivos embarcados, como o ESP32, muito utilizado em aplicações de Internet das Coisas (IoT). Isso a torna uma escolha eficiente para sistemas que exigem baixo custo e eficiência, como automação residencial com reconhecimento facial. A biblioteca permite implementar funcionalidades de visão computacional em dispositivos com recursos limitados, otimizando o uso da memória e do processamento.

Portanto, a OpenCV não é apenas uma ferramenta versátil para visão computacional, mas também um pilar tecnológico no desenvolvimento de soluções modernas, como sistemas de reconhecimento facial. Sua capacidade de lidar com múltiplas etapas do processamento de imagens, combinada com sua facilidade de integração com outras tecnologias, fez dela uma escolha estratégica para o respectivo projeto.

3.2.4 Pyttsx3

A biblioteca "pyttsx3" é uma ferramenta de conversão de texto em fala (Text-to-Speech, TTS) desenvolvida para permitir que aplicações em Python sintetizem voz de maneira eficiente e personalizável. Diferente de outras bibliotecas de TTS que dependem de serviços online, como o gTTS (Google Text-to-Speech), o "pyttsx3" funciona de maneira offline, utilizando os mecanismos de síntese de voz disponíveis no sistema operacional. Isso o torna uma escolha ideal para aplicações que exigem baixa latência e maior controle sobre a conversão de texto em fala, sendo especialmente útil para sistemas embarcados e aplicações que precisam operar sem conexão constante com a internet, justificando o porquê de sua utilização.

No desenvolvimento do sistema de reconhecimento facial para auxiliar pessoas com Alzheimer, a implementação de um sistema de voz foi necessária para tornar a interação mais acessível e intuitiva, uma vez que muitas pessoas que convivem com a doença enfrentam dificuldades na memória e na identificação de familiares e cuidadores. Nesse contexto, um sistema que, além de reconhecer rostos, também fornece informações sonoras sobre a identidade das pessoas identificadas pode ser uma ferramenta extremamente útil para melhorar a autonomia e o bem-estar dos pacientes.

A integração do "pyttsx3" no sistema possibilitou que, ao reconhecer um rosto, o software anunciasse verbalmente o nome da pessoa identificada. Esse processo foi realizado por meio da

extração do nome do indivíduo a partir do banco de dados e da conversão desse texto em voz utilizando a biblioteca. A flexibilidade do "pyttsx3" permitiu ajustes na velocidade, no volume e até mesmo no timbre da voz, tornando a comunicação mais clara e adaptável às necessidades do usuário. Essa personalização é um fator importante, pois algumas pessoas com Alzheimer podem ter maior dificuldade em compreender vozes muito rápidas ou com pronúncia pouco natural.

Outro benefício significativo do uso do "pyttsx3" foi a possibilidade de rodar o sistema de forma local, sem depender de servidores externos para processar a conversão de texto em fala, o que garantiu um tempo de resposta mais rápido, essencial para que a experiência do usuário fosse fluida e natural. Além disso, como o sistema foi projetado para ser utilizado em residências e em ambientes familiares, a operação offline garantiu maior privacidade, evitando que dados sensíveis precisassem ser enviados para a nuvem para processamento.

3.2.5 JSON

Para o sistema de reconhecimento facial funcionar continuamente, o tempo necessário para realizar toda a abordagem seria muito longo. Nesse viés, o JSON é uma solução eficiente para minimizar o tempo de espera para a realização do procedimento.

O JSON (JavaScript Object Notation) é um formato leve de intercâmbio de dados que utiliza uma estrutura de chave-valor para organizar as informações de forma clara e compreensível. Ele é amplamente utilizado devido à sua simplicidade, flexibilidade e capacidade de ser integrado com diversas linguagens de programação, como Python, Java e JavaScript, sendo baseado em texto e tendo como principais vantagens a portabilidade e a fácil manipulação de dados em aplicações que envolvem comunicação entre sistemas distintos.

No projeto de reconhecimento facial, a utilização do JSON foi de grande relevância para realizar melhorias no desempenho do sistema, uma vez que, originalmente, era apresentado um tempo significativo de processamento, em torno de 1 minuto, o que impactava a experiência do usuário. Com o sistema funcionando de forma básica, a atenção foi direcionada para a sua otimização, visando melhorar a eficiência geral.

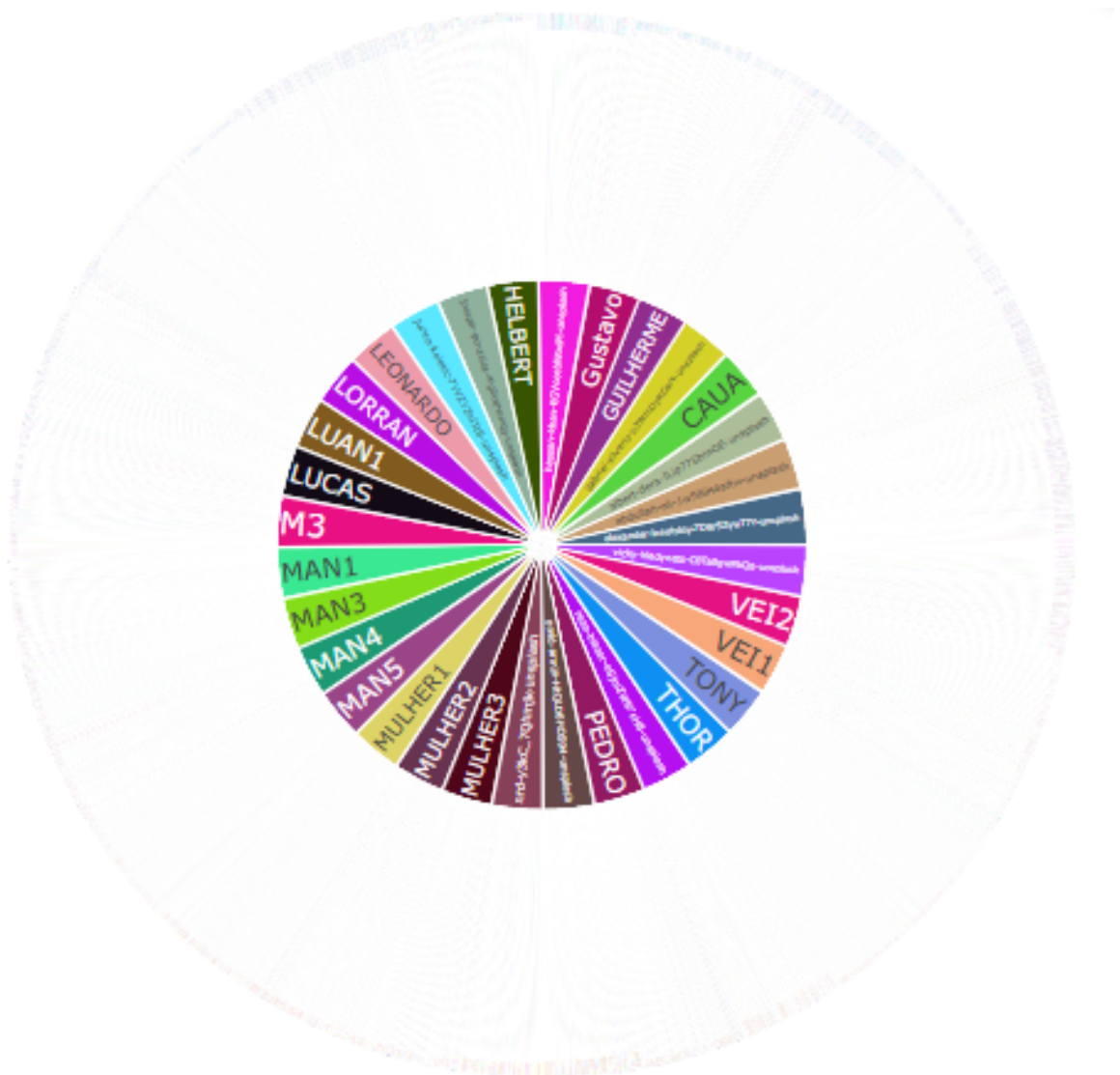
A escolha do JSON como protocolo de comunicação entre o armazenamento de hashes das imagens faciais e a leitura no código em Python foi estratégica, já que, antes disso, a troca de dados entre as diferentes partes do sistema não era tão eficiente, o que contribuía para a lentidão do reconhecimento. Com a implementação do JSON, o formato compacto e estruturado permitiu que os dados fossem transmitidos e processados de maneira muito mais rápida e, dado seu suporte para Python, a integração e manipulação das informações foram simplificadas, auxiliando o sistema como um todo.

A adoção dessa abordagem trouxe resultados favoráveis, reduzindo o tempo total para o reconhecimento facial de 1 minuto para apenas 20 segundos (em casos mais extremos de

espera). Dessa forma, o uso do JSON não apenas otimizou o desempenho técnico do projeto, mas também contribuiu para a experiência geral do usuário, reforçando a importância de escolhas tecnológicas bem planejadas em projetos de automação e IoT.

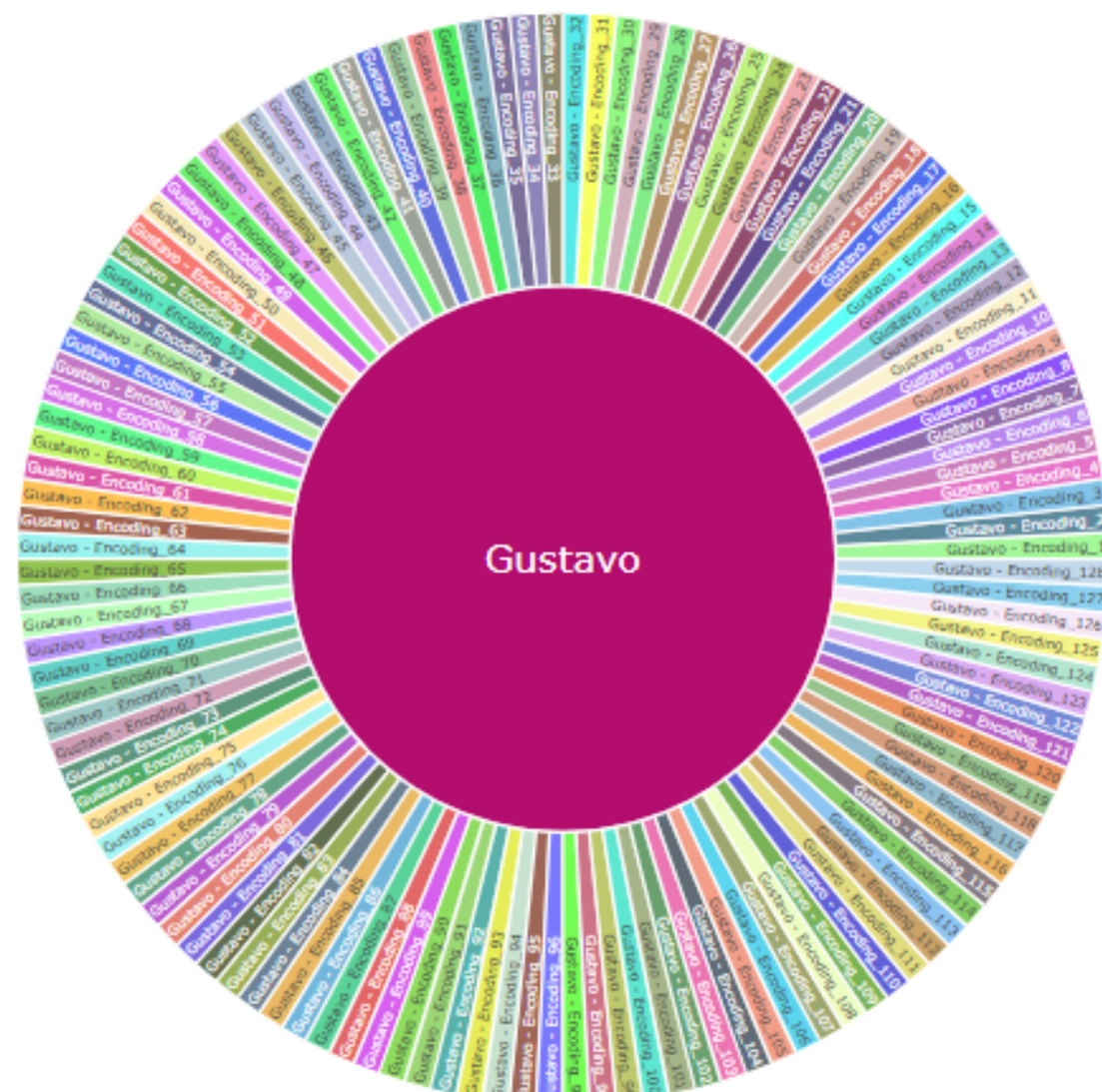
Para incorporá-lo, é necessário carregar um arquivo contendo dois elementos principais, uma lista de nomes e uma lista de codificações faciais associadas, o que é ilustrado nas figuras 8 e 9. Cada codificação facial é representada por um vetor único gerado pela biblioteca que permite identificar um rosto de forma precisa e eficiente. Ao carregar esses dados na memória, o sistema é capaz de realizar comparações em tempo real, eliminando a necessidade de recalcular codificações para cada rosto a cada execução, o que reduz significativamente o tempo de processamento.

Figura 8 – Lista de referências.



Fonte: Autoria Própria

Figura 9 – Codificações associadas a cada nome.



Fonte: Autoria Própria

A integração do JSON no sistema é um dos fatores que tornam esse código eficiente e escalável, já que, em vez de calcular as codificações faciais repetidamente, o sistema armazena esses dados de forma persistente, permitindo que eles sejam reutilizados em várias execuções. Isso não apenas economiza recursos computacionais, mas também melhora significativamente o tempo de resposta do reconhecimento facial. Além disso, o formato JSON facilita a manipulação dos dados, permitindo que novos rostos sejam adicionados ou atualizados de maneira simples, sem necessidade de reestruturar o sistema como um todo.

3.2.6 Aplicação no Código

O código utiliza dos diversos artifícios, citados anteriormente, para realizar o reconhecimento facial. A biblioteca OpenCV é responsável por capturar e processar as imagens da câmera

em tempo real, permite a exibição do vídeo, a detecção das faces e a aplicação de filtros, como o desfoque nas regiões do rosto, o qual varia de 0% a 99%, como parte da proposta de simular condições em que a câmera pode estar suja. Além disso, OpenCV é usada para exibir textos na tela, indicando se uma pessoa foi reconhecida ou não.

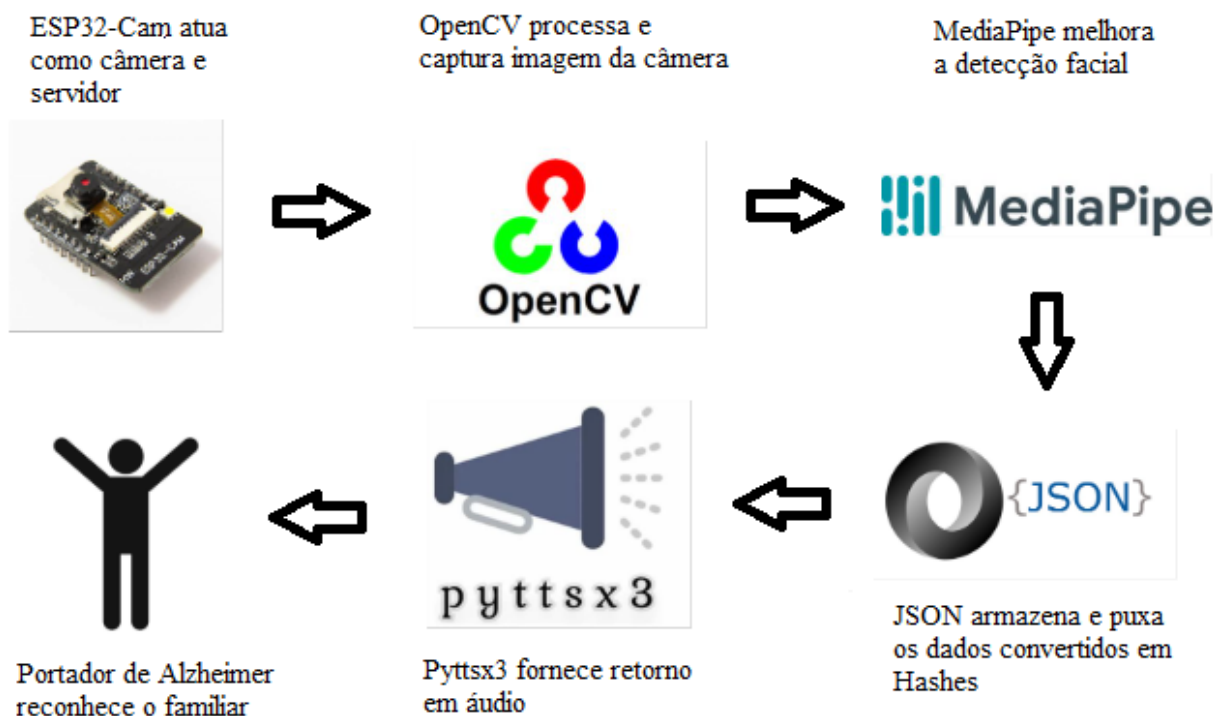
A biblioteca MediaPipe, então, é acionada para melhorar a detecção facial, permitindo uma resposta mais rápida, de forma a auxiliar a biblioteca `face_recognition`, que utiliza modelos de aprendizado de máquina para encontrar e comparar rostos. O código armazena previamente codificações faciais de referência, que são carregadas de um arquivo JSON. Esse arquivo contém um conjunto de dados que associa nomes a representações matemáticas dos rostos, permitindo que o sistema faça comparações entre o rosto capturado e os rostos cadastrados.

O JSON, portanto, desempenha um papel fundamental na estruturação desses dados, garantindo que as informações sobre os usuários sejam armazenadas e recuperadas com facilidade. Ao invés de armazenar as imagens diretamente, o sistema guarda apenas os vetores numéricos que representam os rostos, tornando o processamento mais eficiente. Esse método se relaciona com o conceito de hash, pois cada face pode ser representada por uma assinatura única, facilitando a busca e a comparação entre os dados, já que os dados são armazenados de forma hexadecimal, ocupando menos espaço e permitindo uma interpretação mais rápida da face do indivíduo que está sendo analisado.

Além do reconhecimento visual, o código também utiliza a biblioteca `Pytsx3` para transformar texto em fala, permitindo que o reconhecimento de uma pessoa cadastrada receba uma resposta, em tempo real, com seu nome sendo pronunciado pelo sistema, melhorando a interação com o usuário. Essa funcionalidade adiciona um aspecto mais acessível à aplicação, permitindo que o sistema informe verbalmente sobre quem foi identificado, o que contribui para maior autonomia da pessoa portadora de Alzheimer, já que ela não tem que ser lembrada continuamente por seus familiares ou cuidadores sobre as pessoas que estão ao seu redor. Para evitar repetições desnecessárias, há um controle de tempo que impede que o nome seja falado novamente em intervalos muito curtos.

A combinação dessas tecnologias resulta em um sistema de reconhecimento facial funcional e interativo, capaz de identificar pessoas em tempo real, aplicar efeitos visuais e emitir respostas sonoras, sendo essa correlação expressa pela figura 10.

Figura 10 – Funcionamento das bibliotecas.



Fonte: Autoria Própria

Outro aspecto crucial foi o código desenvolvido para o ESP32-CAM, com o objetivo de capturar imagens por meio da câmera conectada ao microcontrolador, processá-las e transmiti-las para um servidor via conexão WiFi. Para isso, ele inicia importando bibliotecas essenciais que possibilitam a interação com a câmera, a conexão com a rede e a manipulação das imagens capturadas, além de definir os pinos utilizados pelo módulo ESP32-CAM para estabelecer a comunicação entre o hardware e o software, de modo a garantir o correto funcionamento da captura de imagens.

Além da configuração da câmera, o código determina parâmetros importantes para a transmissão de dados, como o formato das imagens, que são enviadas no padrão JPEG, e posteriormente convertidas em Hashes, bem como a estrutura do fluxo de transmissão via HTTP. Um servidor é implementado para permitir que os dispositivos conectados acessem a transmissão em tempo real e, para evitar sobrecarga de memória, os recursos utilizados são liberados ao final de cada transmissão.

Outro aspecto relevante do código é a função setup, que realiza as configurações iniciais do sistema. Durante essa etapa, o ESP32 desativa recursos de proteção para evitar interferências no funcionamento da câmera e estabelece parâmetros como a taxa de transmissão do monitor

serial e, em seguida, configura a câmera definindo os pinos responsáveis pela comunicação com o microcontrolador, o tamanho das imagens capturadas e a qualidade da compressão JPEG. Essas configurações equilibram a qualidade das imagens transmitidas com a eficiência na utilização de recursos do ESP32.

Por fim, o código inicia a conexão com a rede WiFi utilizando as credenciais fornecidas, sendo exibidas mensagens, as quais são exibidas no monitor serial indicando o status da conexão. Após a conexão bem-sucedida, o endereço IP do ESP32 é exibido, permitindo que os usuários acessem a transmissão de vídeo por meio de um navegador, possibilitando, dessa maneira, a captura e o envio contínuo de imagens, criando uma integração eficiente entre o hardware do ESP32-CAM e o restante do sistema de reconhecimento facial.

3.3 Montagem do produto

A montagem do esquemático do sistema foi realizada com o objetivo de combinar a funcionalidade de reconhecimento facial com um dispositivo simples, composto por um óculos convencional e uma unidade embarcada para processamento. Para isso, o primeiro passo foi a escolha dos componentes necessários, sendo a câmera do ESP32, responsável pela captura de imagens da face das pessoas em tempo real. Esse dispositivo foi conectado a um módulo de processamento embarcado que, por meio dos algoritmos previamente citados, ficou responsável por realizar o reconhecimento facial.

A base de dados foi configurada para armazenar as imagens das pessoas cujos rostos seriam reconhecidos e ajustada pelo sistema de hashes para proporcionar maior velocidade e menor utilização de espaço, o que possibilitou ao sistema identificar com rapidez e precisão os indivíduos que aparecessem no campo de visão da câmera. O software de processamento foi ajustado para realizar essa identificação em tempo real, reduzindo o tempo de resposta entre a captura da imagem e a reação do sistema.

Além disso, foi necessário configurar o sistema de voz, que foi integrado de maneira que, ao identificar um rosto no banco de dados, fosse acionado um comando de áudio correspondente, sendo alimentado por um módulo de síntese vocal projetado para dizer o nome da pessoa identificada. A comunicação entre os diferentes módulos foi facilitada por uma interface de programação, garantindo a fluidez da operação do sistema como um todo.

A montagem física dos componentes envolve a integração da câmera e do display de modo que ficassem confortáveis nos óculos, exigindo precisão. A escolha do formato e do design foi fundamental para que o sistema fosse usável, sem comprometer o conforto do usuário. Foram realizados vários testes, tanto no hardware quanto no software, para garantir que todos os componentes funcionassem corretamente juntos, sendo o sistema final representado pela figura 11.

Figura 11 – Esquemático do sistema.



Fonte: Autoria Própria

3.4 Experimento e Análise dos dados

O objetivo central da etapa de experimento e análises foi explorar o comportamento de um sistema ou processo em função de três variáveis principais: o nível de desfoque (Blur), a tolerância e a inclinação. Para isso, foram realizados ensaios controlados, nos quais essas variáveis foram sistematicamente manipuladas, e os resultados foram registrados de forma estruturada.

O desfoque (Blur) foi variado em diferentes níveis, partindo de 0 até 99, com incrementos regulares. A tolerância, por sua vez, foi ajustada em intervalos que variaram de 0.10 a 1.00, abrangendo uma ampla gama de condições operacionais. A inclinação foi testada em três ângulos distintos: 0°, 45° e 90°, permitindo avaliar o impacto dessa variável em diferentes orientações. Para cada combinação desses parâmetros, foram realizados quatro ensaios, nos quais foram medidos o tempo de execução e o acerto em cada tentativa. Esses dados foram então consolidados, calculando-se a média e o desvio padrão dos tempos de execução, bem como a acurácia, que representa a proporção de acertos em relação ao total de ensaios.

Para complementar a análise, foi construída uma tabela de correlação, que permitiu explorar as relações entre as variáveis estudadas. A matriz de correlação é uma ferramenta estatística utilizada para medir a relação entre variáveis dentro de um conjunto de dados, sendo representada por uma matriz quadrada onde cada elemento indica o coeficiente de correlação entre duas variáveis. O coeficiente de correlação mais comumente utilizado varia de -1 a 1, onde valores próximos de 1 indicam uma correlação positiva forte, ou seja, quando uma variável aumenta, a outra tende a aumentar também. Valores próximos de -1 indicam uma correlação negativa forte, significando que o aumento de uma variável está associado à diminuição da outra, enquanto valores próximos de 0 sugerem que não há uma relação linear significativa entre as variáveis.

Essa matriz pode ser visualizada por meio de mapas de calor, de modo a facilitar a in-

terpretação dos dados ao destacar as correlações mais significativas com diferentes intensidades de cores, o que propicia uma percepção mais rápida de quais variáveis estão mais relacionadas, possibilitando deduções sobre fenômenos complexos.

Outro critério aplicado para análise estatística foi o Gráfico de Pareto, uma ferramenta amplamente utilizada para análise de impacto de variáveis em processos e tomada de decisão que permite identificar os fatores mais relevantes dentro de um conjunto de variáveis analisadas. O critério de corte desse tipo de gráfico, em 80%, se baseia no princípio de que aproximadamente 80% dos efeitos advêm de 20% das causas. Na prática, isso significa que ao focar nos fatores responsáveis pelos primeiros 80% do impacto acumulado, é possível otimizar significativamente um sistema sem a necessidade de intervir em todas as variáveis, o que foi abordado nos resultados.

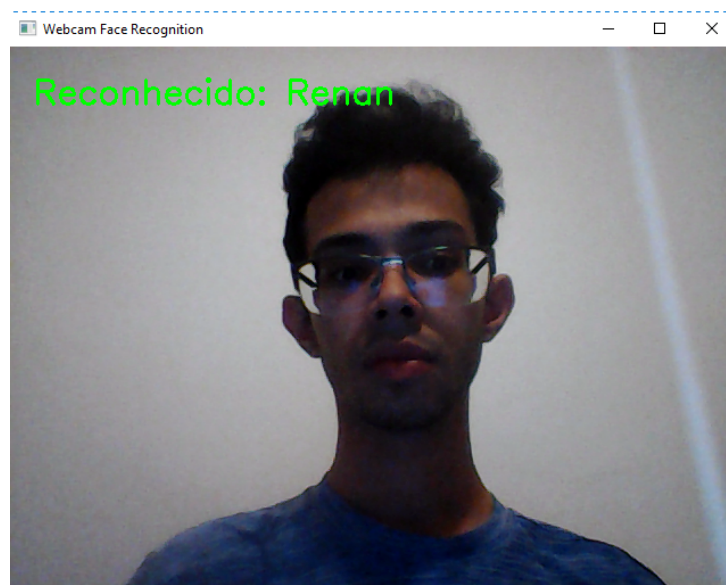
4 RESULTADOS

Com o desenvolvimento do projeto, foi possível realizar uma série de testes abrangentes, os quais permitiram compreender seu comportamento em diferentes cenários e condições de uso. Ademais, a análise dos resultados possibilitou a identificação das principais vantagens oferecidas pela solução, como sua acessibilidade, eficiência e integração com outras tecnologias. Por outro lado, também foi possível mapear suas limitações, incluindo eventuais restrições técnicas, desafios operacionais e aspectos que podem ser aprimorados em futuras versões do projeto.

4.1 Reconhecimento de pessoas distintas

Os testes iniciais exploratórios foram realizados para avaliar a capacidade do sistema de reconhecimento facial em identificar diferentes pessoas foram conduzidos com o objetivo de validar a eficácia e a confiabilidade da solução proposta, especialmente no contexto de auxílio a pacientes com Alzheimer. Esses testes foram projetados para simular situações reais de uso, considerando variáveis como condições de iluminação, ângulos faciais, expressões e a presença de indivíduos com características faciais semelhantes, como demonstrado pela Figura 12, em que o teste obteve sucesso apesar de ter grau de parentesco (irmão). A importância desses testes reside no fato de que o sistema deve ser capaz de distinguir com precisão entre diferentes indivíduos, uma vez que a identificação correta é fundamental para garantir a segurança e o bem-estar dos usuários. Em um cenário onde o sistema será utilizado por pessoas com Alzheimer, a capacidade de reconhecer familiares, cuidadores ou mesmo o próprio paciente é de suma importância para evitar situações de confusão ou desorientação, que são comuns nessa condição.

Figura 12 – Teste de parentesco.



Fonte: Autoria Própria

A capacidade do sistema de distinguir entre diferentes pessoas foi particularmente importante em situações em que a identificação rápida e precisa pode prevenir situações de risco, como a perda de um paciente em um local público ou a confusão entre cuidadores e estranhos. Além disso, o sistema mostrou-se eficaz ao lidar com imagens de baixa resolução ou parcialmente obstruídas, o que é comum em ambientes reais onde a captura de imagem pode não ser ideal.

Em suma, os testes exploratórios realizados comprovaram que o sistema é capaz de identificar diferentes pessoas com precisão, mesmo em condições adversas. Essa capacidade de distinção garante que o sistema possa ser utilizado de forma confiável no dia a dia de pacientes com Alzheimer e seus cuidadores.

4.2 Sistema de Voz

O teste de voz avaliou a capacidade da biblioteca "pyttsx3" em gerar uma comunicação compreensível, com foco especial na pronúncia de nomes próprios, que frequentemente apresentam desafios devido à variedade de origens linguísticas e fonéticas. Para isso, foram selecionados nomes de diferentes origens, incluindo nomes comuns em línguas como português, inglês e espanhol, como "Gustavo", "Man1", "Juan" e outras, a fim de verificar a adaptabilidade da ferramenta. A pronúncia foi realizada corretamente, tendo nomes com "J" definidos em espanhol com a fonética de "R", e nomes em português e inglês sendo expressados tal qual em seus respectivos países, assim provando sua versatilidade.

Um dos aspectos mais notáveis do teste foi a fluidez da comunicação. A pyttsx3 mostrou-se capaz de gerar uma fala natural, sem interrupções ou pausas inadequadas, o que contribuiu para uma experiência auditiva agradável e eficiente. Além disso, o sistema de voz mostrou-se eficiente em termos de desempenho, processando rapidamente as solicitações de síntese de voz sem comprometer a qualidade do áudio gerado. Isso foi particularmente importante em cenários onde a velocidade de resposta é crítica, como no reconhecimento facial para auxiliar pessoas com Alzheimer.

Em conclusão, o teste de voz utilizando a biblioteca pyttsx3 se mostrou eficiente, demonstrando que a ferramenta é capaz de fornecer uma comunicação fluída e precisa, especialmente no que diz respeito à pronúncia de nomes próprios. A combinação de fluidez, precisão e desempenho faz da pyttsx3 uma escolha adequada para aplicações que demandam síntese de voz de alta qualidade.

4.3 Níveis de tolerância

O nível de tolerância é um parâmetro significativo, que define a sensibilidade do sistema ao identificar uma face, influenciando diretamente o tempo de reconhecimento e a acurácia da aplicação. Em termos técnicos, o nível de tolerância está relacionado ao valor de comparação

das distâncias entre as codificações faciais, sendo que, quanto menor for esse valor, mais rígido o sistema se torna na identificação de faces. Dessa maneira, para que o sistema reconheça uma pessoa como correspondente a uma face armazenada no banco de dados, a distância calculada entre as codificações faciais da pessoa identificada e a referência deve ser inferior ao nível de tolerância especificado, sendo esse conceito importante para garantir que o sistema não faça correspondências incorretas ou falsas positivas, o que é crítico em contextos como o cuidado de pacientes com Alzheimer, onde uma falha de reconhecimento pode ter impactos negativos.

A função de tolerância foi responsável por ajustar a relação entre falsos positivos (faces não correspondentes erroneamente identificadas como correspondentes) e falsos negativos (faces correspondentes não identificadas corretamente) e, a partir dessa métrica de comparação, o sistema decide se deve ou não acionar a resposta de reconhecimento, o que implica em uma resposta de voz.

Os resultados obtidos variaram significativamente dependendo do nível de tolerância utilizado. Os testes iniciais, realizados com níveis de tolerância entre 0,1 e 0,5 (que representam 10% e 50% de grau de semelhança com a foto da base de dados), não apresentaram sucesso, falhando consistentemente em reconhecer as faces e ultrapassando o tempo máximo de espera, que foi definido como 10 segundos para os testes, o que pode ser explicado pela rigidez excessiva desses valores. Esses níveis muito baixos de tolerância fizeram com que o sistema se tornasse extremamente seletivo, não permitindo sequer uma pequena margem de erro entre as codificações faciais, o que gerou uma taxa muito alta de falsos negativos, onde o sistema não conseguiu identificar corretamente pessoas que, de fato, estavam na base de dados de referência, levando à falha do reconhecimento.

Por outro lado, ao aumentar drasticamente o nível de tolerância para 0,6, embora o sistema tenha sido mais flexível, houve uma quantidade considerável de falsos positivos e a precisão do reconhecimento foi mais baixa em comparação com níveis mais altos de tolerância. Esse foi o pior dos níveis que conseguiram algum sucesso, o que sugere que um nível de tolerância muito alto pode ser problemático ao ponto de permitir correspondências incorretas entre faces, gerando uma baixa confiabilidade na aplicação.

O nível de tolerância de 0,7 se apresentou como o ponto ideal para o sistema. Esse valor foi o mais eficaz, resultando no melhor desempenho geral, com uma boa taxa de acertos e um número razoável de falsos positivos e negativos. Esse valor permitiu que o sistema reconhecesse com precisão as faces sem ser excessivamente rigoroso, ao mesmo tempo em que minimizava o risco de falsas identificações.

Embora os níveis de tolerância entre 0,8 e 1 também apresentassem resultados positivos, com o sistema sendo capaz de identificar corretamente as faces em diversas ocasiões, 0,7 se destacou como superior. Isso pode ser atribuído ao fato de que níveis mais altos de tolerância tornam o sistema mais permissivo, o que pode introduzir um número maior de falsos positivos e diminuir a precisão global. Embora o sistema tenha sido capaz de fazer algumas correspondências

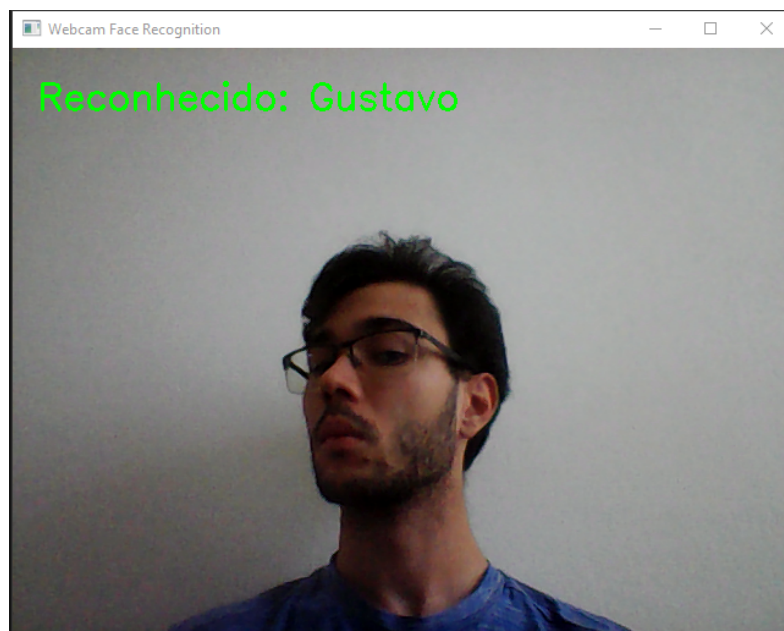
corretas, a flexibilidade adicional comprometeu a confiança nos resultados.

Apesar dos valores se mostrarem extremamente rígidos quanto à necessidade de semelhança com a foto registrada no banco de dados, isso é normal nesse tipo de aplicação, já que modelos de reconhecimento facial ainda não são tão aprimorados. Isso é evidenciado pelo trabalho de (Pratik *et al.*, 2023) em que, mesmo apresentando um modelo mais robusto, a taxa de acertos só se mostrou favorável, alcançando 99,38% de precisão, a partir da definição de um nível de tolerância de 0,5.

4.4 Angulação

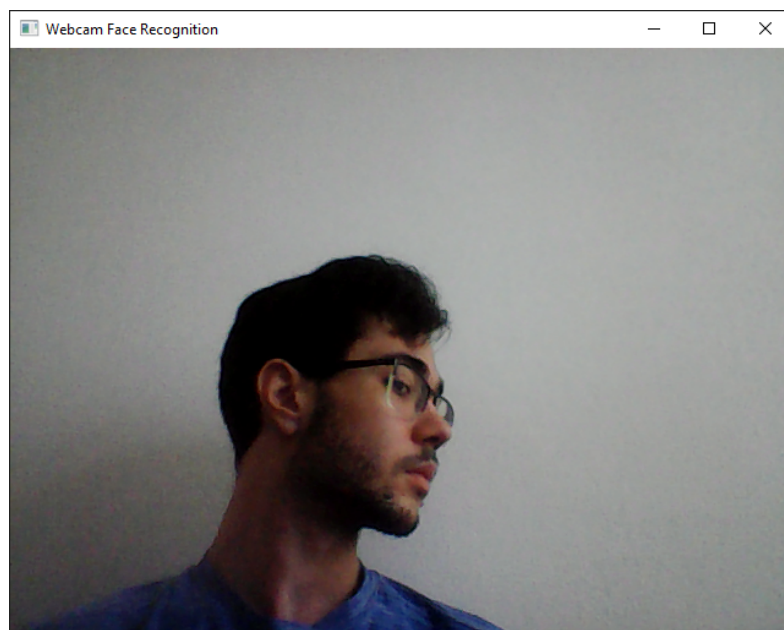
Outro fator crucial a ser abordado diz respeito ao ângulo em que o rosto é reconhecido, dado que o posicionamento do reconhecimento varia de acordo com a localização do usuário e o modo como o rosto é interpretado. Nesse viés, avaliou-se a inclinação de três modos distintos, com 0° representando o rosto de frente, 45° o rosto inclinado e 90° com o rosto totalmente de lado, tal como demonstrado pela Figura 13 e Figura 14.

Figura 13 – Rosto 45°.



Fonte: Autoria Própria

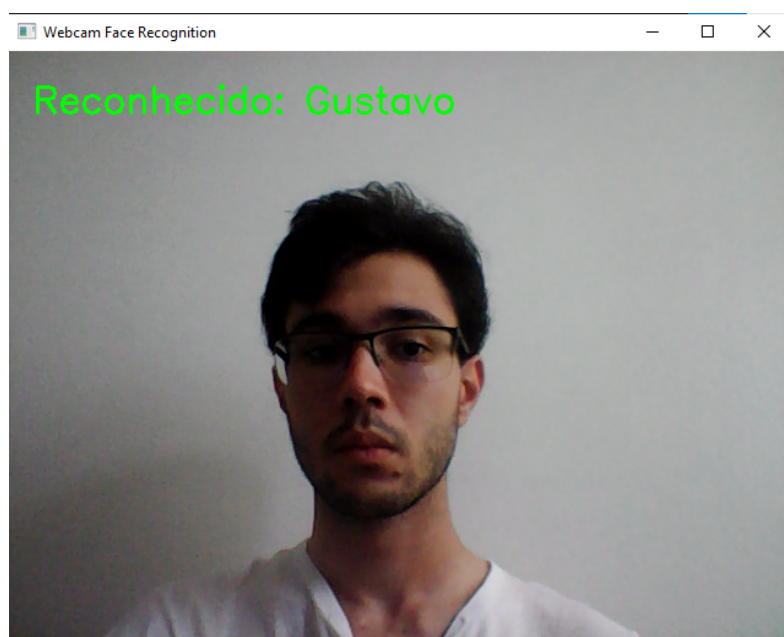
Figura 14 – Rosto 90°.



Fonte: Autoria Própria

Para imagens capturadas com inclinação de 0°, os resultados foram altamente positivos, com altas taxas de acurácia e reconhecimento bem-sucedido. Isso sugere que o algoritmo está bem ajustado para rostos posicionados diretamente de frente para a câmera, garantindo um desempenho confiável em cenários ideais, tal como ilustrado pela Figura 15.

Figura 15 – Rosto 0°.



Fonte: Autoria Própria

Entretanto, quando a inclinação foi aumentada para 45°, começaram a surgir falhas no reconhecimento, já que a taxa de acurácia registrou falhas em alguns testes, mostrando resultados inconsistentes em certas condições. Esse comportamento indica que o algoritmo encontra dificuldades na identificação de feições quando há uma inclinação moderada, o que pode comprometer a usabilidade do sistema em algumas situações reais em que os usuários podem não estar perfeitamente alinhados com a câmera.

Nos testes conduzidos com uma inclinação de 90°, os resultados foram completamente negativos, não havendo nenhuma tentativa de reconhecimento bem-sucedida, demonstrando que o sistema não é capaz de identificar corretamente rostos posicionados lateralmente. Isso era esperado, já que o algoritmo utilizado não possui mecanismos avançados de normalização facial que permitam o reconhecimento sob grandes variações angulares, uma vez que a biblioteca `face_recognition` detecta pontos-chave do rosto, como olhos, nariz e boca, e transforma essas coordenadas em um vetor numérico. Nesse contexto, como grandes inclinações não possibilitam realizar uma clara identificação de tais pontos, falhas de identificação são extremamente suscetíveis, o que é normal.

4.5 Teste de luminosidade

Com o intuito de averiguar as limitações de reconhecimento da câmera utilizada, o teste de luminosidade envolveu diferentes condições de iluminação a fim de verificar sua capacidade de identificação em ambientes variados.

De maneira a simular condições de diferentes luminosidades, providenciou-se um ambiente completamente escuro, sendo a intensidade de luz gradualmente variada por meio do acréscimo de outras lanternas.

Ao realizar o experimento inicialmente com o local totalmente escuro, constatou-se que, em condições de extrema escuridão em que a face do usuário se torna invisível para a câmera, o sistema apresenta limitações significativas, tornando-se incapaz de realizar o reconhecimento corretamente. Esse resultado era esperado, visto que os algoritmos utilizados dependem de informações visuais para a detecção e identificação facial.

Com a presença de uma luz artificial dessa vez, o algoritmo se mostrou eficiente, conseguindo reconhecer o usuário apesar da não prevalência de uma fonte natural de luminosidade, como demonstrado pela Figura 16. Isso é crucial, dado que, diversas vezes, os usuários poderão se encontrar em situações de rotina em que há pouca presença de luz natural, tornando importante o reconhecimento apesar das limitações dessa fonte de iluminação.

Figura 16 – Teste com uma fonte de luz.



Fonte: Autoria Própria

Por fim, a quantidade de fontes de luz foi aumentada para duas, além de haver uma aproximação das mesmas em relação à câmera, de maneira a simular situações em que há presença de intensa luz artificial, como abordado na Figura 17. Apesar da mudança, o sistema também se mostrou confiável na identificação, a despeito das adversidades, não sendo um fator responsável por aplicar instabilidade no projeto sob condições normais.

Figura 17 – Teste com duas fontes de luz.



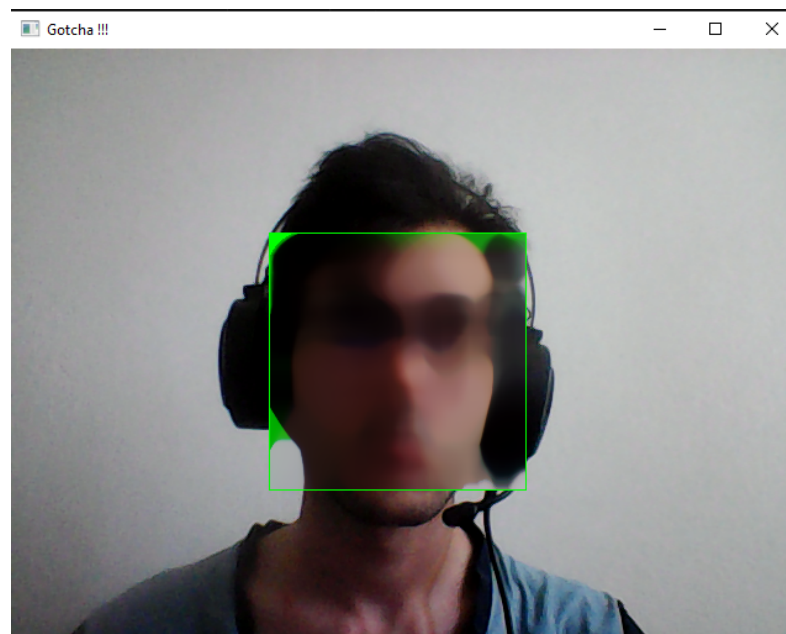
Fonte: Autoria Própria

Dessa forma, o mínimo de luz artificial já permite a identificação se os pontos-chave forem detectados com sucesso. No entanto, a dependência da presença de luminosidade para a identificação facial é um fator limitante que pode ser abordado em trabalhos futuros, buscando aperfeiçoar ainda mais a tecnologia aplicada.

4.6 Teste de blur

Para avaliar a robustez do sistema de reconhecimento facial em condições adversas, foi realizado um teste variando o nível de blur(desfoque) aplicado à imagem da câmera, ilustrado pela Figura 18. O teste consistiu em aumentar progressivamente o desfoque em valores de 0, 20, 40, 60, 80 e 99, simulando situações em que a lente estivesse suja ou embaçada devido a condições climáticas. Esse experimento é especialmente relevante para o projeto, pois o portador pode não perceber que a câmera está obstruída por sujeira, umidade ou outros fatores, comprometendo a eficácia do reconhecimento facial.

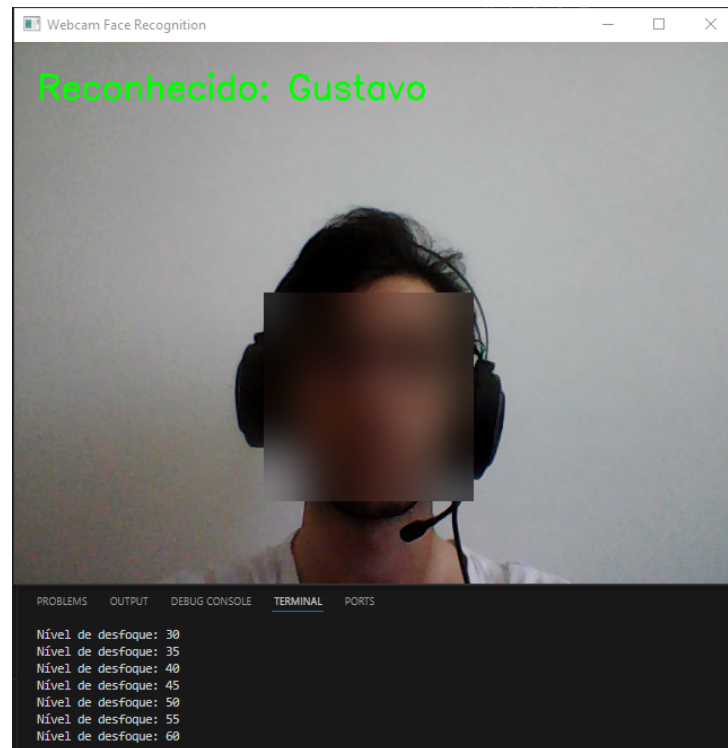
Figura 18 – Teste de blur.



Fonte: Autoria Própria

Os resultados demonstraram que o sistema manteve um desempenho satisfatório nos níveis de blur 0%, 20%, 40% e 60%, conseguindo identificar corretamente os rostos capturados, como ilustrado pela figura 19.

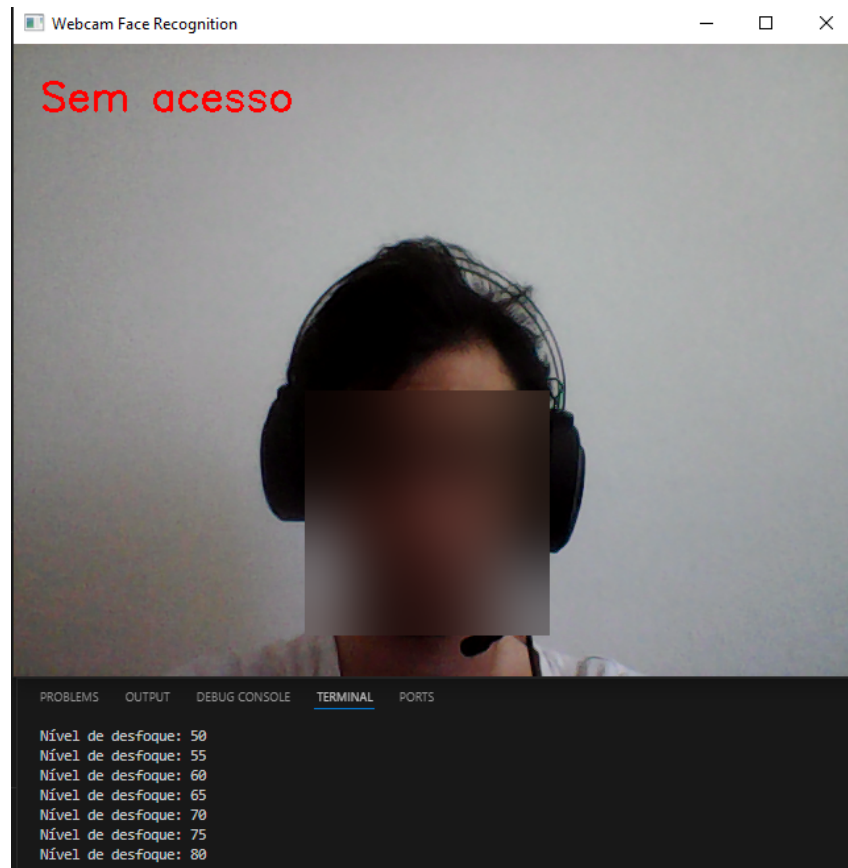
Figura 19 – Desfoque de 0 a 60 (%).



Fonte: Autoria Própria

No entanto, em níveis mais elevados, especificamente em 80 e 99, o reconhecimento teve falhas, indicando que altos graus de desfoque impactam a precisão do sistema, como visto na figura 20. Dessa forma, conclui-se que o sistema apresenta um desempenho confiável sob condições normais e moderadamente adversas, mas sua precisão é comprometida em situações de desfoque severo.

Figura 20 – Desfoque de 80 a 99 (%).



Fonte: Autoria Própria

4.7 Análise dos Dados

Ao incorporar diferentes valores de variáveis em condições distintas, foi necessário montar uma tabela para organizar os dados e identificar quais fatores possuíam maior influência, assim como os menos relevantes. A Tabela 21 ilustra alguns valores dos experimentos realizados, evidenciando essa organização.

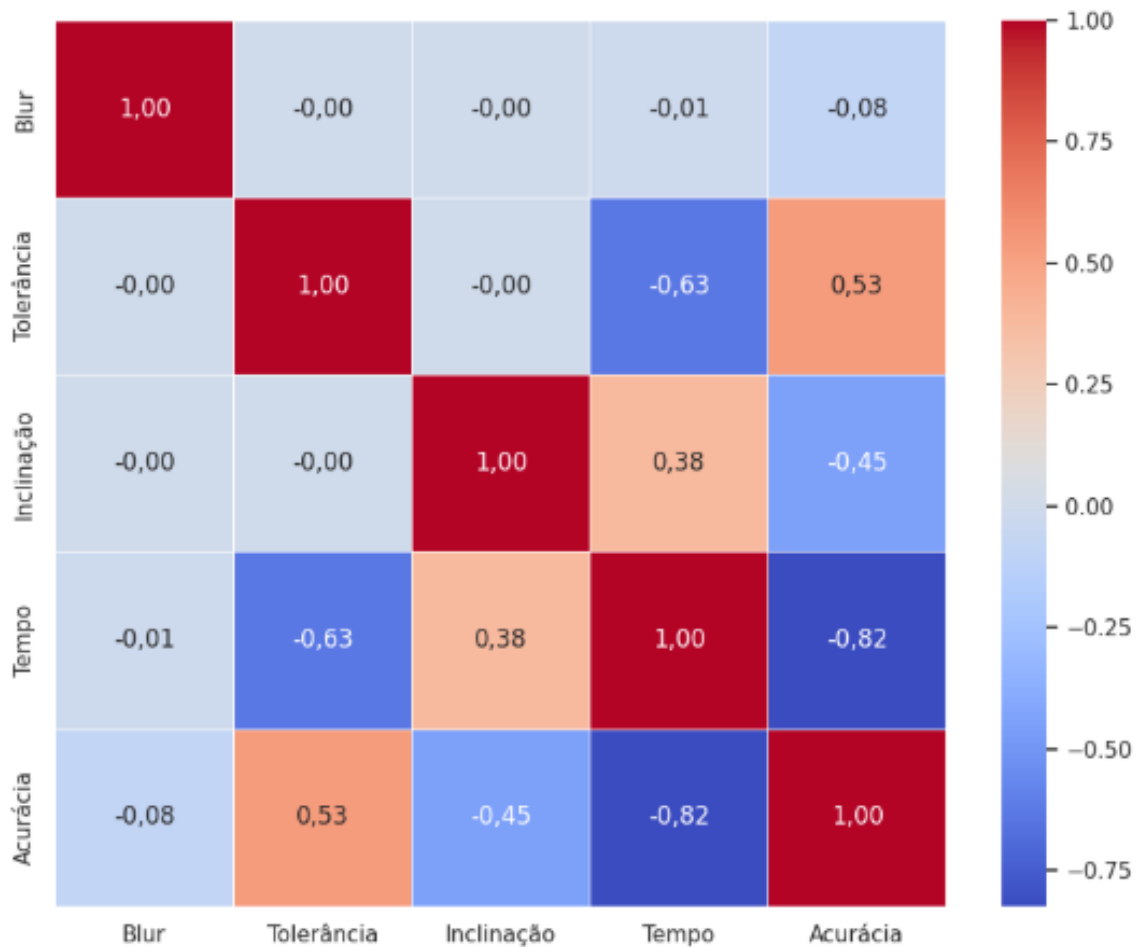
Figura 21 – Exemplos da Tabela de Dados.

| | Blur | Tolerância | Inclinação | Tempo 1 | Tempo 2 | Tempo 3 | Tempo 4 | Média | Desvio | Acerto 1 | Acerto 2 | Acerto 3 | Acerto 4 | Acurácia |
|-----|------|------------|------------|---------|---------|---------|---------|-------|--------|----------|----------|----------|----------|----------|
| 0 | 0.0 | 0.1 | 0.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 10 | 60.0 | 0.1 | 45.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 20 | 0.0 | 0.2 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 30 | 80.0 | 0.2 | 0.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 40 | 20.0 | 0.3 | 45.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 50 | 80.0 | 0.3 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 60 | 40.0 | 0.4 | 0.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 70 | 99.0 | 0.4 | 45.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 80 | 40.0 | 0.5 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 90 | 0.0 | 0.6 | 0.0 | 3.25 | 10.00 | 10.00 | 6.87 | 10.00 | 0.00 | 1.0 | 0.0 | 0.0 | 1.0 | 0.50 |
| 100 | 60.0 | 0.6 | 45.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 110 | 0.0 | 0.7 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 120 | 80.0 | 0.7 | 0.0 | 3.54 | 2.81 | 2.77 | 2.98 | 3.03 | 0.36 | 0.0 | 1.0 | 1.0 | 1.0 | 0.75 |
| 130 | 20.0 | 0.8 | 45.0 | 2.56 | 2.44 | 2.50 | 2.51 | 2.50 | 0.05 | 1.0 | 1.0 | 1.0 | 0.0 | 0.75 |
| 140 | 80.0 | 0.8 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 150 | 40.0 | 0.9 | 0.0 | 1.71 | 2.04 | 1.72 | 1.72 | 1.80 | 0.16 | 1.0 | 1.0 | 1.0 | 0.0 | 0.75 |
| 160 | 99.0 | 0.9 | 45.0 | 1.97 | 1.89 | 1.54 | 2.05 | 1.86 | 0.22 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |
| 170 | 40.0 | 1.0 | 90.0 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 0.00 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 |

Fonte: Autoria Própria

A fim de assimilar a interferência entre os diferentes critérios utilizados, utilizou-se da tabela de correlação para deduzir os testes mais impactantes para a taxa de acertos, bem como o bom funcionamento do projeto, tal como demonstrado pela Figura. 22.

Figura 22 – Matriz de correlação.

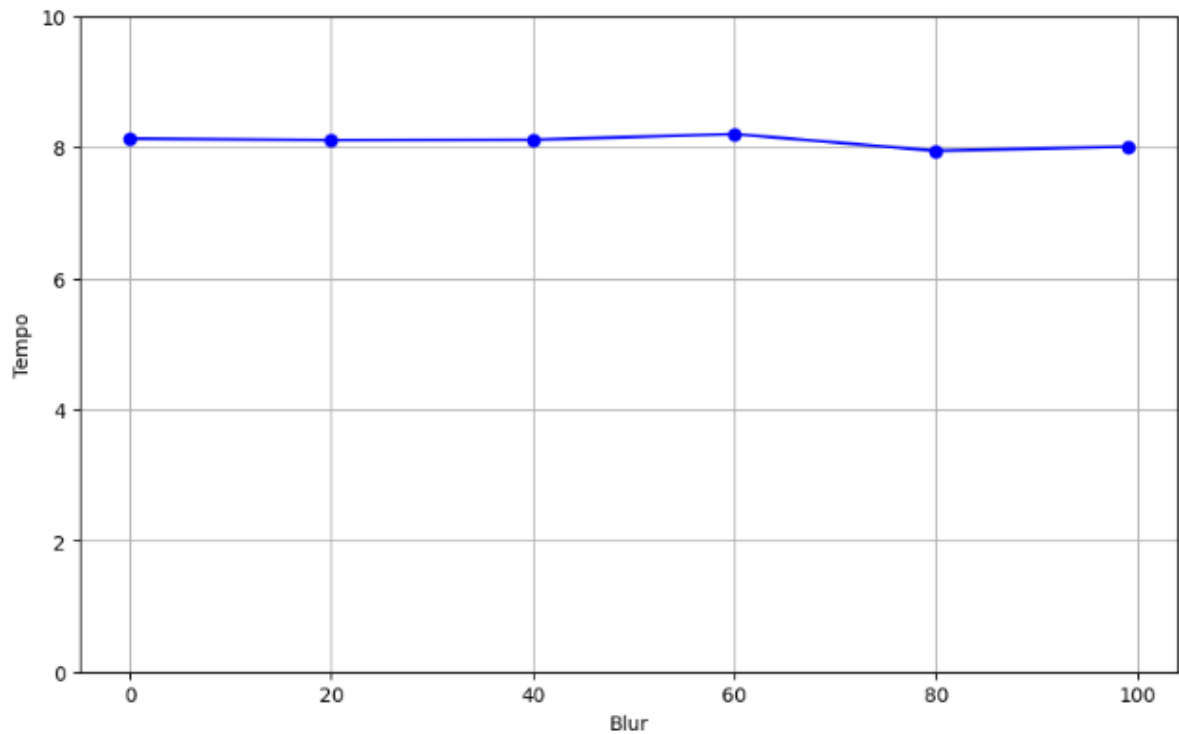


Fonte: Autoria Própria

A matriz de correlação revelou padrões importantes sobre o desempenho do modelo de reconhecimento facial em diferentes condições de teste. O desfoque da imagem (Blur) apresenta correlação reduzida com os tempos de resposta, sugerindo que, no geral, sua presença não compromete drasticamente o desempenho do modelo. No entanto, observações específicas dos testes 80 e 99 indicam que o desfoque foi mais impactante nesses casos, podendo sugerir que certos níveis de blur ultrapassam a capacidade do modelo de realizar um reconhecimento preciso. Apesar disso, a ausência de uma correlação significativa na matriz geral pode indicar que o modelo conseguiu lidar com níveis mais leves de desfoque sem grande prejuízo ao tempo de reconhecimento.

A relação entre o desfoque e tempo de reconhecimento fica ainda mais visível a partir da representação gráfica demonstrada pela Figura 24. Os resultados sugerem que, inicialmente, o blur não contribuiu com um maior tempo de resposta do sistema. No entanto, em níveis mais elevados, esse mesmo tempo sofreu, o que se deve à menor quantidade de informação sendo analisada pela câmera. Isso comprova que, no geral, a correlação entre esses fatores é baixa, tal como demonstrado pela matriz de correlação.

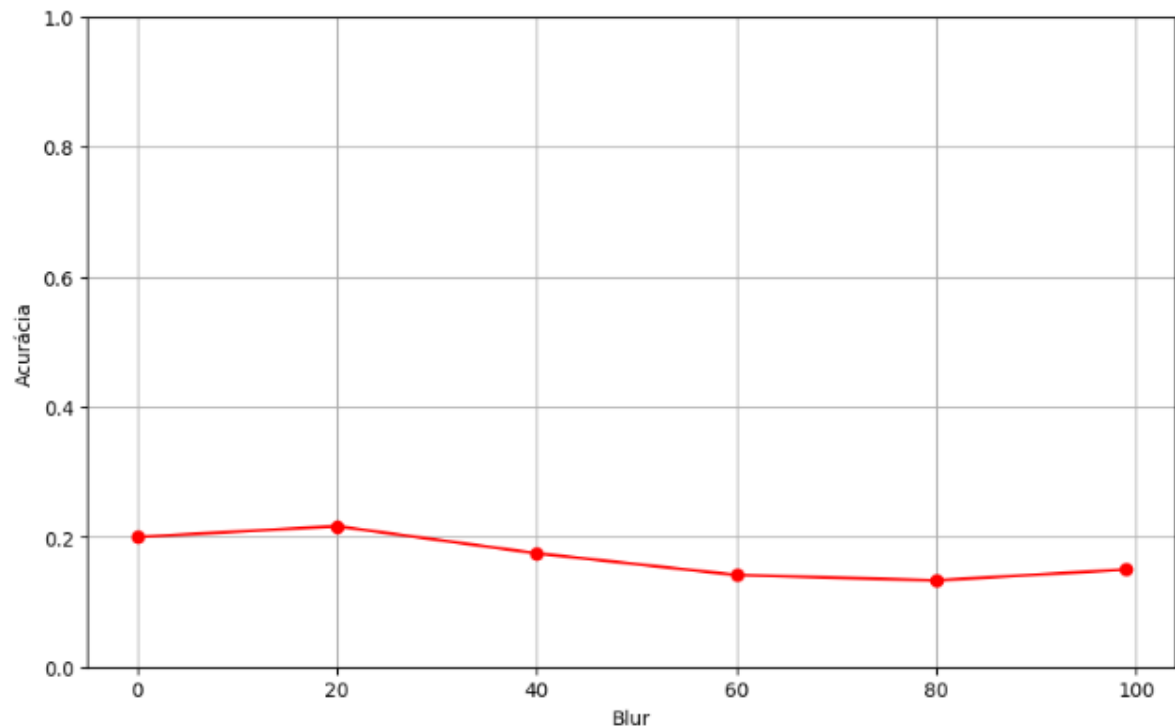
Figura 23 – Gráfico Blur x Tempo de Reconhecimento.



Fonte: Autoria Própria

A variável blur (desfoque), ao ser analisada graficamente com relação à acurácia, apresenta o mesmo padrão, mostrando não ter grande impacto conforme sua variação, tal como apontado pela Figura 24. Isso demonstra que o sistema se comporta de forma semelhante, apesar de diferentes graus de blur, embora ainda seja recomendada a limpeza periódica do dispositivo a fim de minimizar as adversidades relacionadas ao desfoque.

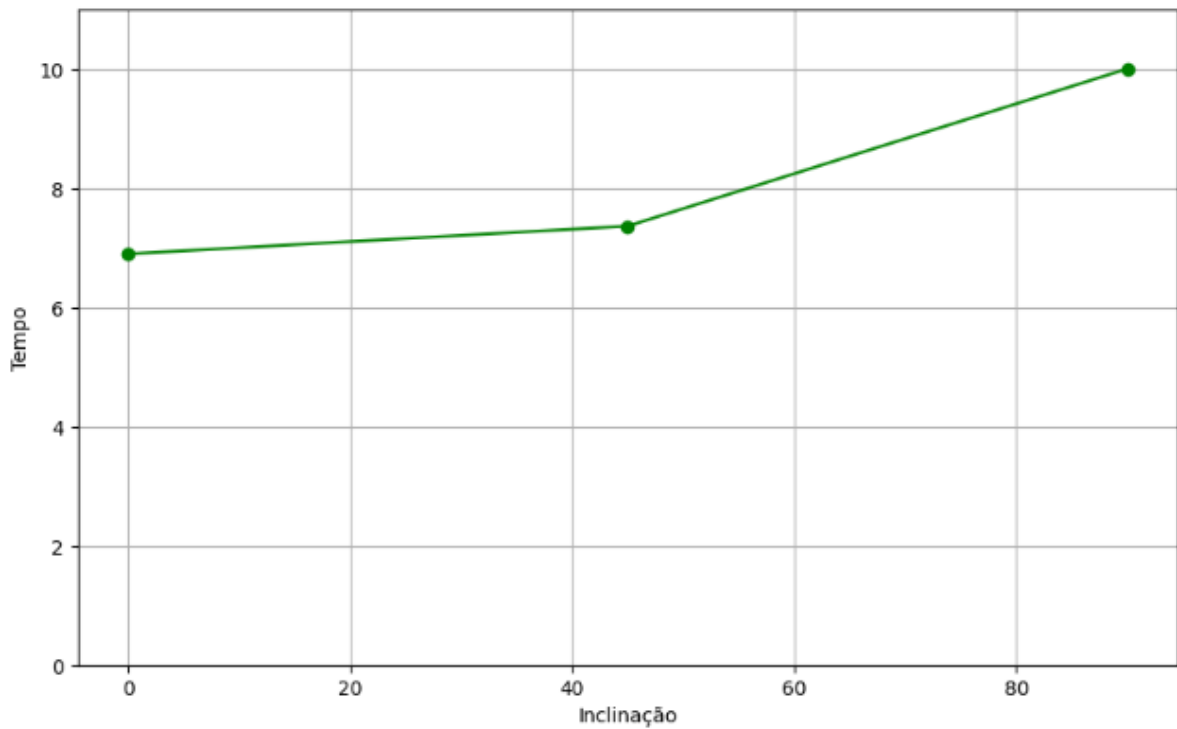
Figura 24 – Tabela Blur x Acurácia.



Fonte: Autoria Própria

A inclinação da face demonstra uma correlação moderada com os tempos de reconhecimento, variando entre 0,38 e 0,39, demonstrando que a posição do rosto impacta o tempo necessário para identificação, mas não de maneira tão determinante. O modelo parece ser relativamente robusto a pequenas variações na inclinação, embora sua performance ainda possa ser otimizada para cenários com maior variação de posicionamento facial, o que é visível na Figura 25.

Figura 25 – Gráfico Inclinação x Tempo de Reconhecimento.

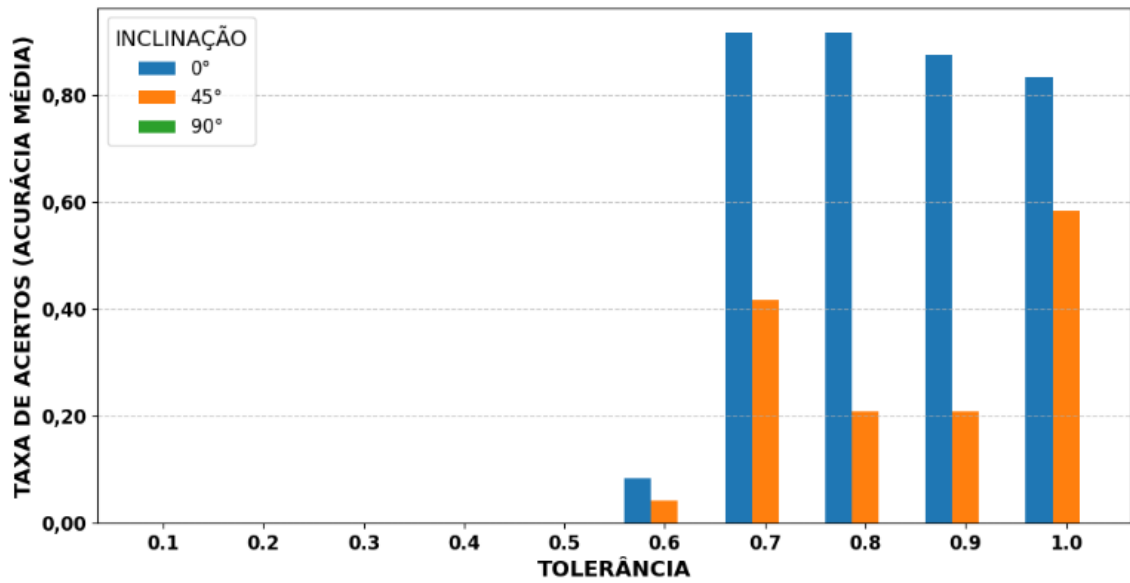


Fonte: Autoria Própria

O tempo de reconhecimento se mostra crescente conforme o aumento da inclinação, apresentando resultados melhores para 0° , onde é possibilitado um melhor reconhecimento dos pontos-chave da face da pessoa identificada, justificando o desempenho. Enquanto isso, a inclinação de 45° apresenta uma leve variação, indicando que o sistema pode apresentar certa demora para reconhecer rostos curvados e, para 90° , o tempo se manteve constante em 10 segundos, dado que não houve acertos, ultrapassando o limite do período de espera.

A inclinação também demonstrou impactar de forma negativa na taxa de acertos registrados, variando de -0,37 a -0,41, já que, como abordado previamente, o modelo é extremamente dependente do reconhecimento de pontos-chave. Nesse viés, o ângulo de 90° teve grande interferência na taxa de acertos, tal qual o de 45° sob certas circunstâncias, como evidenciado pelo gráfico da Figura 26.

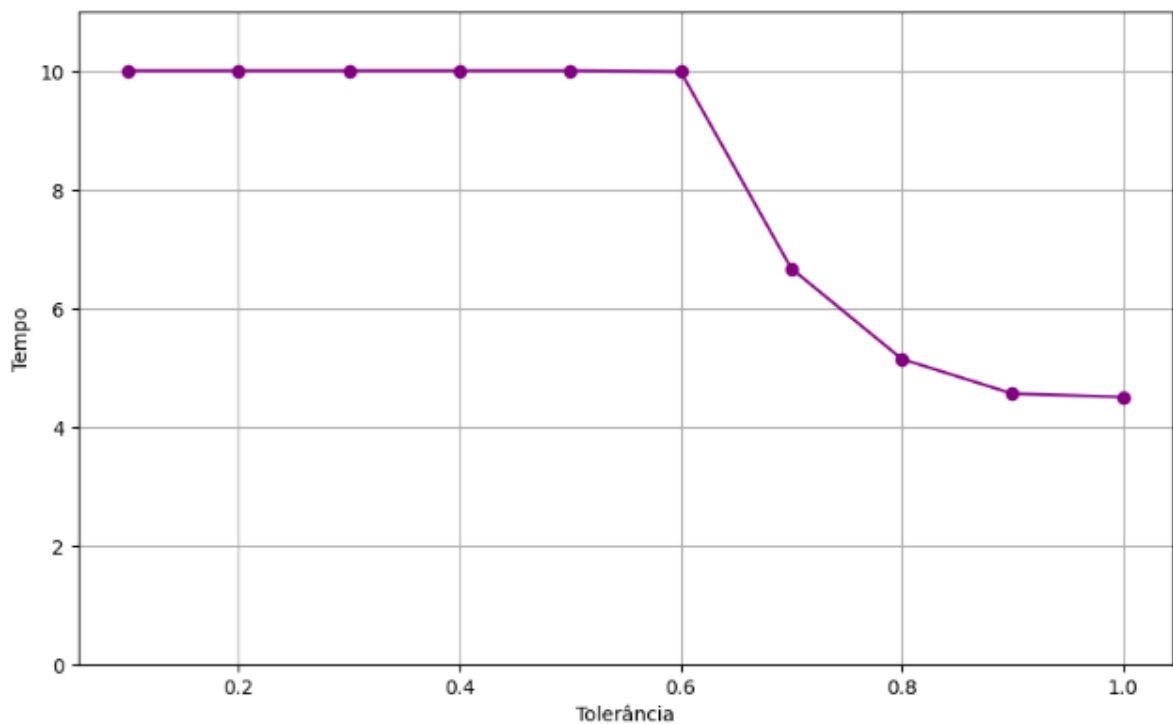
Figura 26 – Gráfico Acurácia x Inclinação.



Fonte: Autoria Própria

A tolerância do sistema foi um fator importante para o sucesso do reconhecimento facial. A correlação de -0,63 com o tempo de resposta indica a capacidade desse parâmetro impactar negativamente o tempo de reconhecimento caso mal ajustado, comprometendo a eficiência do sistema, como visto na Figura 27.

Figura 27 – Gráfico Graus de Tolerância X Tempo de Reconhecimento.

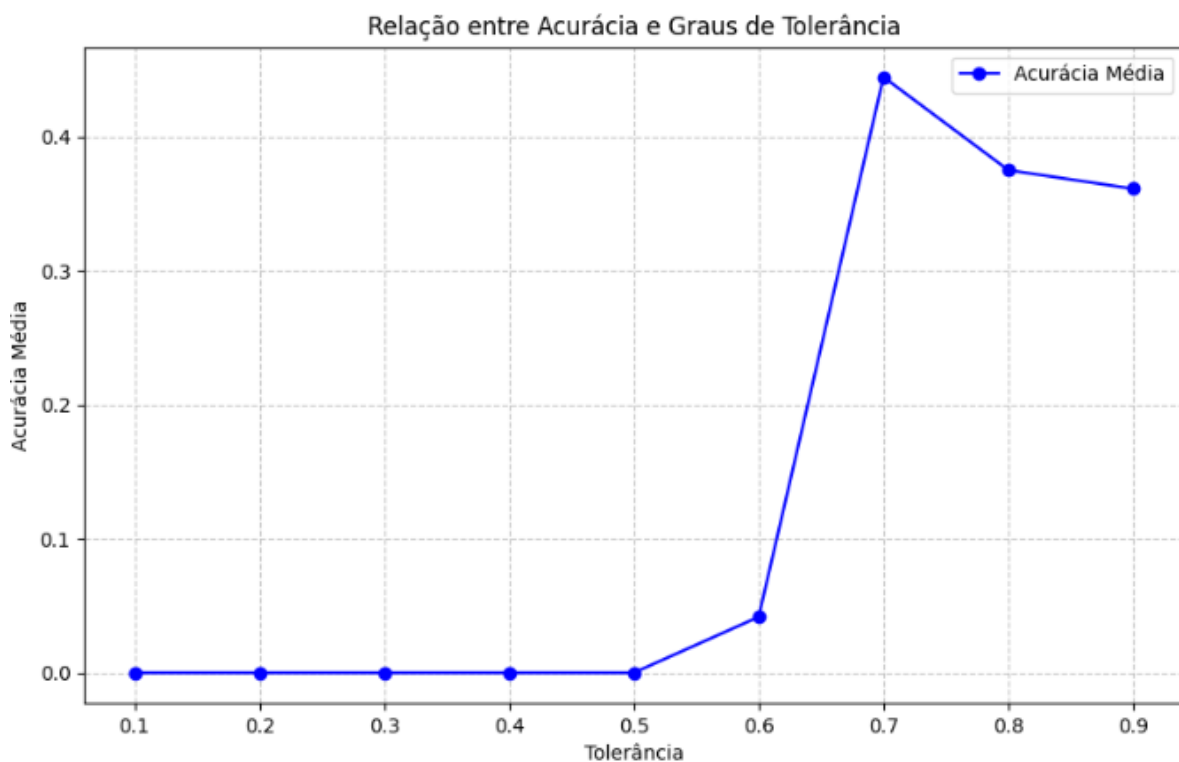


Fonte: Autoria Própria

Durante os testes, falhas no reconhecimento ocorreram quando a tolerância estava entre 0,1 e 0,5, sugerindo que valores baixos restringem excessivamente a margem de variação para a identificação da face, ocasionando o tempo máximo de espera de 10 segundos. A partir de 0,6, os acertos começaram a ocorrer e o tempo diminuiu, havendo menor duração de espera quanto maior o grau de tolerância, já que a associação exata com a foto arquivada no banco de dados reduziu.

O impacto do grau de tolerância na acurácia também foi notável, sendo 0,6 o pior grau de desempenho, dentro daqueles que apresentaram acertos, e 0,7 o melhor, reforçando que encontrar um equilíbrio adequado na tolerância é essencial para otimizar o tempo de resposta e minimizar erros de reconhecimento, o que é visível na Figura 28.

Figura 28 – Gráfico Acurácia x Graus de Tolerância.



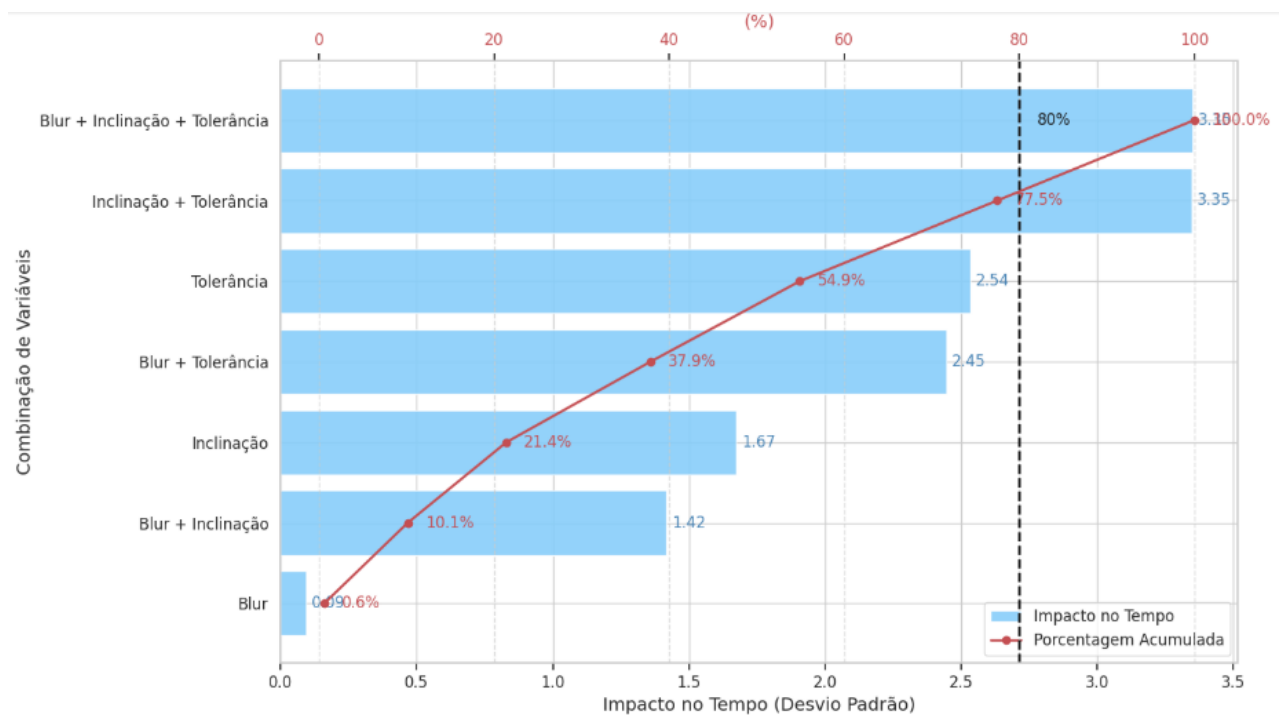
Fonte: Autoria Própria

A análise dos dados evidencia a importância de compreender as relações entre os fatores que afetam o reconhecimento facial. A forte relação entre os tempos de resposta confirma a estabilidade do modelo entre os testes, enquanto a influência moderada da inclinação e do desfoque em testes específicos destaca possíveis áreas para melhorias. A relação entre tolerância e acertos sugere que ajustes nesse parâmetro podem otimizar significativamente o desempenho do sistema, garantindo maior precisão e velocidade no reconhecimento facial.

4.8 Análise Estatística

Para uma verificação mais aprofundada dos resultados, contruiu-se um gráfico de Pareto, o qual revelou padrões sobre a influência da associação das diferentes variáveis nas respectivas saídas: tempo de reconhecimento e acurácia. A Figura 29 explora o impacto desses padrões, focando-se no tempo de reconhecimento.

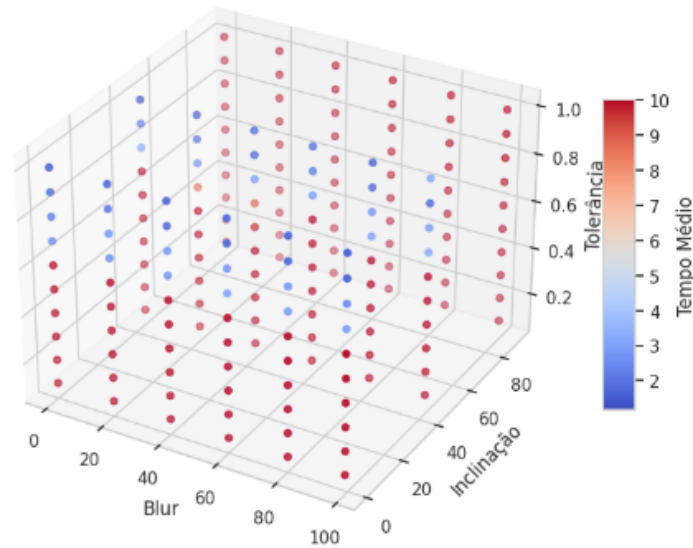
Figura 29 – Gráfico de Pareto do Tempo de Reconhecimento.



Fonte: Autoria Própria

O principal fator identificado é a combinação “Blur + Inclinação + Tolerância”, que representa a variável com o maior efeito no tempo de reconhecimento, destacando-se como um dos principais pontos críticos a serem considerados. Isso sugere que, quando esses três fatores estão combinados, o impacto no tempo aumenta consideravelmente, indicando que há uma interação significativa entre eles, já que dificultam significativamente a capacidade do sistema reconhecer os pontos-chave, o que é descrito na Figura 30.

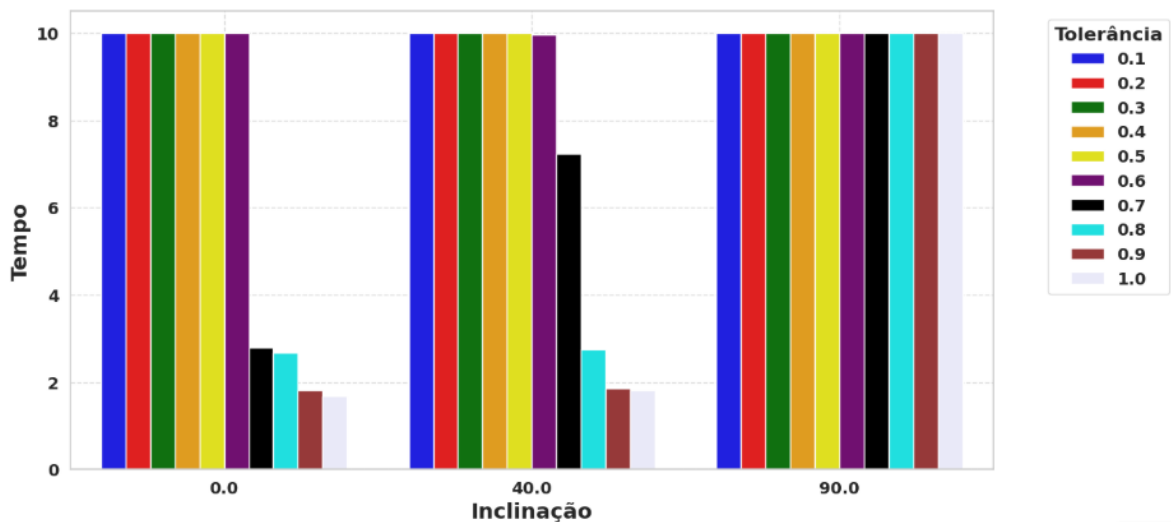
Figura 30 – Gráfico de Interação: Blur x Tolerância x Inclinação no tempo de reconhecimento.



Fonte: Autoria Própria

A segunda variável mais impactante, “Inclinação + Tolerância”, reforça essa ideia, mostrando que mesmo na ausência de desfoque, a combinação desses dois fatores ainda representa uma influência considerável no tempo de reconhecimento, o que também aponta que o blur não possui um impacto tão grande individualmente (dado o leve deslocamento no gráfico), tal como demonstrado pela Figura 31. Isso indica que a tolerância aos erros e a inclinação das imagens afetam diretamente a precisão do sistema, exigindo mais processamento para corrigir ou interpretar corretamente os dados. Já a variável “Tolerância”, isolada, surge como o terceiro maior fator de impacto, demonstrando que, independentemente de outros aspectos, ajustes nos limites de tolerância são determinantes para o desempenho do sistema.

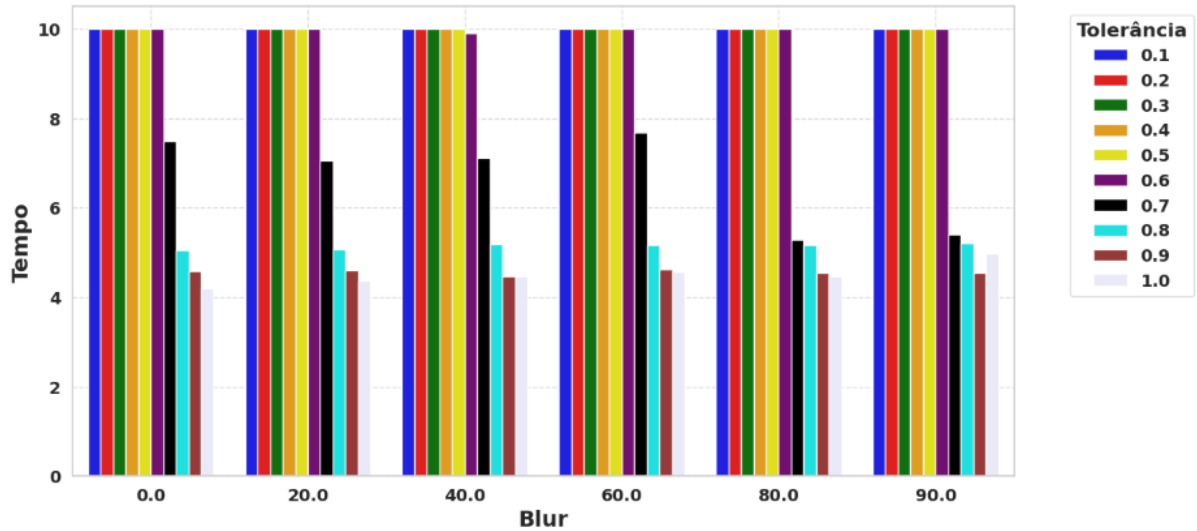
Figura 31 – Gráfico de Interação: Tolerância x Inclinação no tempo de reconhecimento.



Fonte: Autoria Própria

A variável “Blur + Tolerância” aparece em sequência, se mostrando menos impactante que o grau de tolerância sozinho para aumento do tempo de reconhecimento. Isso sugere que o blur é capaz de minimizar a quantidade de informação a ser processada pelo sistema, o que agiliza o processo em algumas ocasiões, o que é visto na Figura 32.

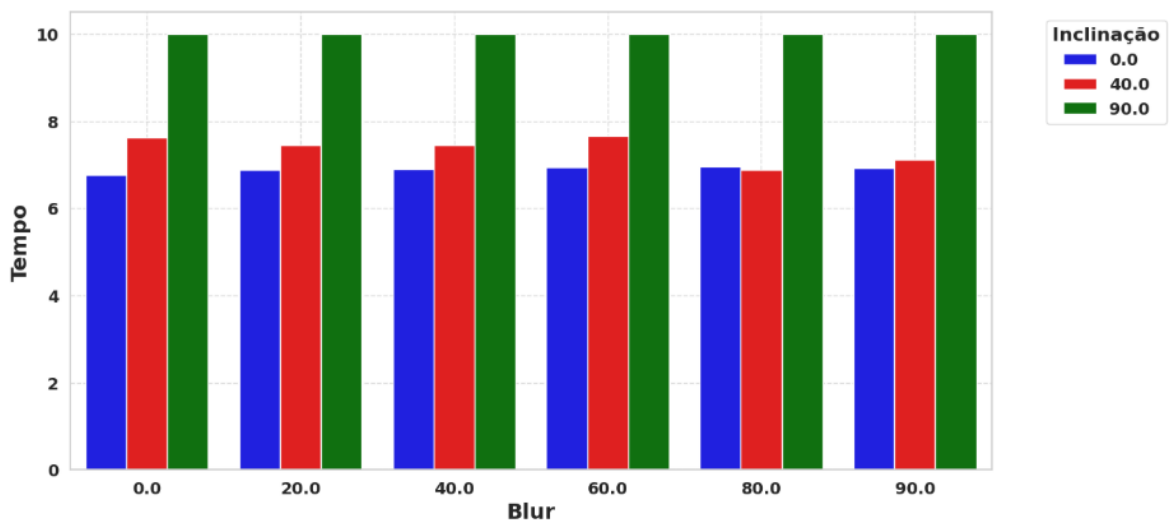
Figura 32 – Gráfico de Interação: Tolerância x Blur no tempo de reconhecimento.



Fonte: Autoria Própria

A inclinação, quando analisada isoladamente, apresenta um impacto menor em comparação com as combinações anteriores, o que indica que, embora relevante, seu efeito isolado pode ser mais gerenciável do que o grau de tolerância. No entanto, seus resultados foram mais expressivos do que quando combinada junto ao blur, que pode ter impactado positivamente pela menor quantidade de informação a ser analisada, sendo essa relação descrita na Figura 33.

Figura 33 – Gráfico de Interação: Blur x Inclinação no tempo de reconhecimento.



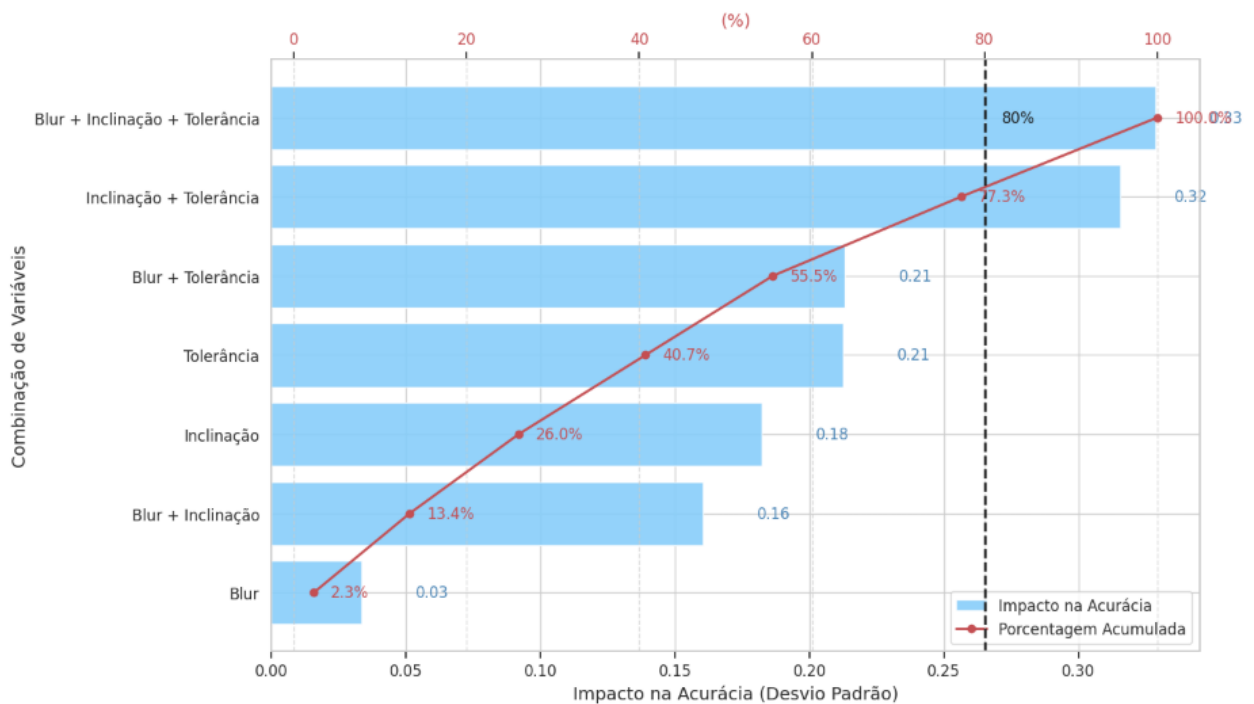
Fonte: Autoria Própria

Por fim, a variável menos impactante do gráfico é o “Blur” isolado, indicando que o desfoque, por si só, não representa um grande desafio para o tempo de processamento quando não está combinado com outros fatores.

Essas observações mostram que os fatores que mais impactam o tempo do reconhecimento facial não agem isoladamente, mas sim em conjunto. O corte em 80% evidencia que o foco deve ser dado às principais combinações, pois são elas que mais influenciam no desempenho do sistema. Assim, estratégias de otimização devem priorizar a redução dos efeitos negativos dessas combinações críticas, como ajustes na tolerância para compensar a inclinação e a presença de desfoque, melhorando a eficiência geral do processo.

Quando analisado o impacto dessas variáveis na acurácia, certos padrões também podem ser notados, como evidenciado na Figura 34.

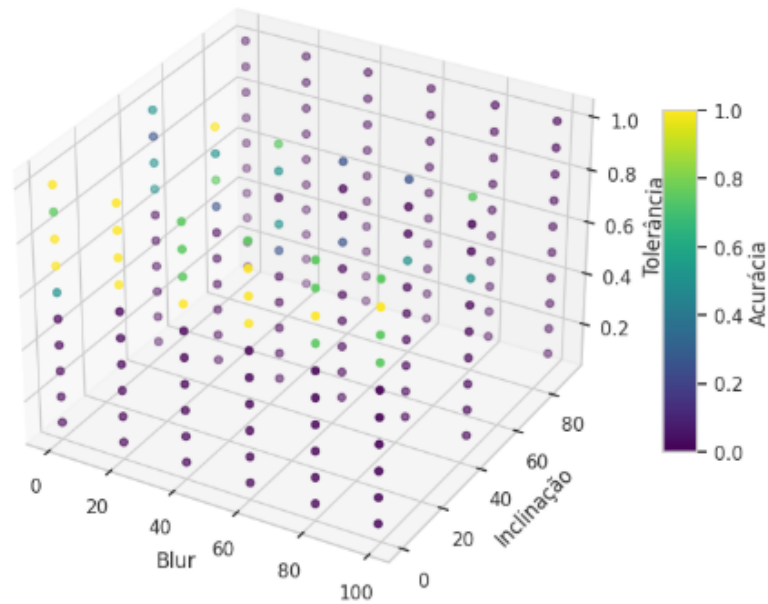
Figura 34 – Gráfico de Pareto da Acurácia.



Fonte: Autoria Própria

A combinação "Blur + Inclinação + Tolerância" representa a interação completa entre as três variáveis, sendo o desfoque menos impactante, como mencionado previamente, mas tendo seu efeito mais perceptível quando combinado com a inclinação e a tolerância. A grande dificuldade de reconhecimento de pontos-chave justifica a junção desses fatores ser tão impactante, sendo isso representado no gráfico da Figura 35.

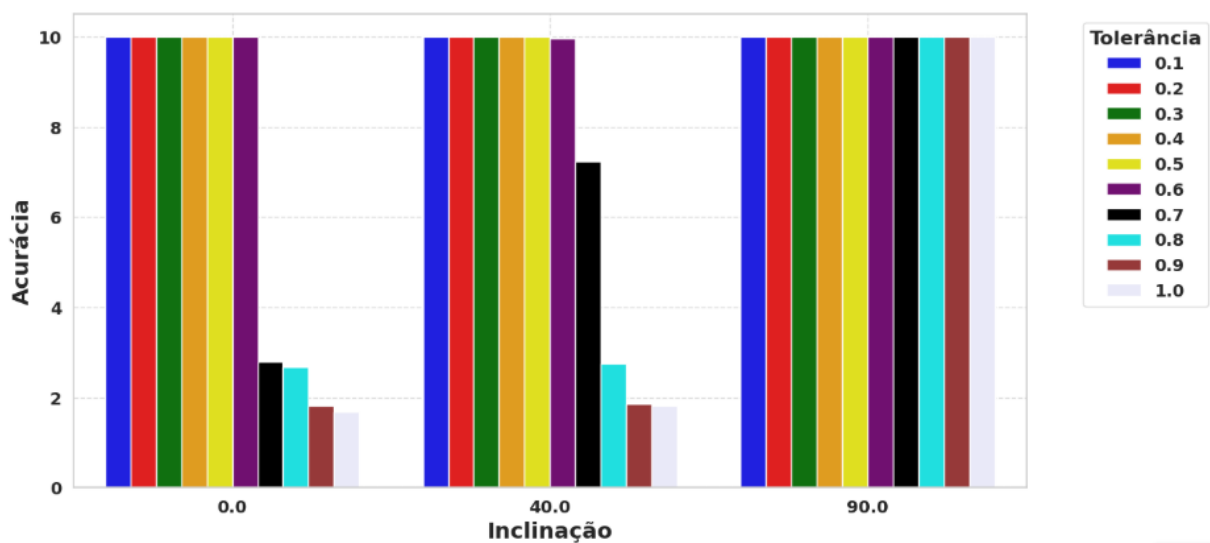
Figura 35 – Gráfico de Interação: Tolerância x Inclinação na acurácia.



Fonte: Autoria Própria

A combinação "Inclinação + Tolerância" exclui o Blur, mas ainda assim apresenta um impacto considerável. As inclinações de 45° e 90° são fatores críticos, pois afetam diretamente a capacidade do sistema de identificar pontos-chave no rosto e, quando combinadas com graus de tolerância de 0,1 a 0,6, os erros se tornam extremamente suscetíveis. Isso é expressado na Figura 36

Figura 36 – Gráfico de Interação: Tolerância x Inclinação na acurácia.

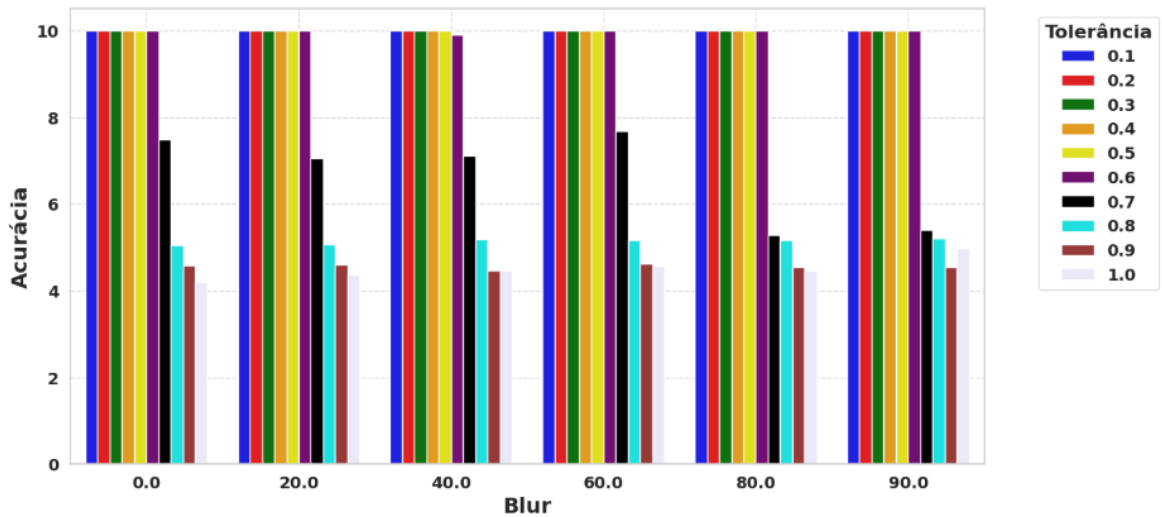


Fonte: Autoria Própria

A combinação "Blur + Tolerância" mostra que, embora o desfoque tenha um impacto menor, ele ainda contribui para a variabilidade da acurácia quando combinado com outros fatores,

como visto no gráfico da Figura 37, além de ressaltar o impacto que a tolerância tem para uma identificação bem-sucedida ser realizada. Isso se torna extremamente visível quando tal variável, mesmo analisada individualmente, se mostra à frente da combinação das outras duas variáveis, ainda que abaixo do fator de corte.

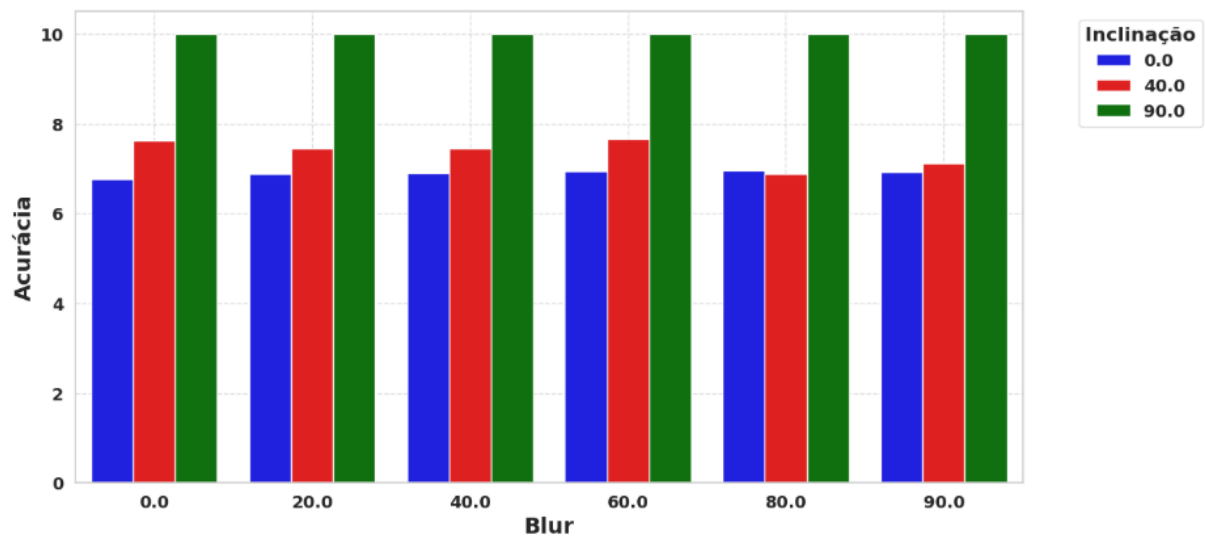
Figura 37 – Gráfico de Interação: Tolerância x Blur na acurácia.



Fonte: Autoria Própria

Em sequência, o mesmo fenômeno é notado, a tolerância se mostra como mais impactante na acurácia, seguido da inclinação, tal qual ocorrido no gráfico de tempo de reconhecimento. A combinação "Blur + Inclinação", que tem sua relação mostrada na Figura 38, apresenta a segunda menor interferência na acurácia e, por último, tem-se o blur isolado, podendo ser inferido que, durante os testes, as amostras com desfoque não fizeram tanta diferença, mas ainda apresentaram resultados discrepantes em certos experimentos.

Figura 38 – Gráfico de Interação: Blur x Inclinação na acurácia.



Fonte: Autoria Própria

4.9 Custos e viabilidade

A escolha dos componentes considerou tanto a viabilidade técnica quanto os custos envolvidos, o que pode ser visualizado na Tabela 1.

| Produto | Preço (R\$) | Quantidade |
|--|---------------|------------|
| Óculos sem Lente | 99,00 | 1 |
| Esp32 Cam Node MCU (MOD1) | 34,59 | 1 |
| 3-Watt Speaker | 6,29 | 1 |
| MAX98357A I2S 3-Watt Class D Amplifier | 7,39 | 1 |
| 5V USB Power Supply Cable | 6,89 | 1 |
| Custo Total | 154,16 | |

Tabela 1 – Tabela de Custos dos Produtos (Cotação realizada em 07/01/2025)

Um dos itens principais foi o "Óculos sem Lente", que, ao custo de 99 reais, serviu como a estrutura física do dispositivo. É válido destacar que o preço da armação do óculos pode variar, dependendo do modelo escolhido. Essa variação de preço é um critério de gosto do usuário e pode ser ajustada conforme a necessidade, já que diferentes tipos de armações estão disponíveis no mercado.

Outros componentes importantes foram o Esp32 Cam Node MCU (MOD1), com custo de 34,59 reais, que atuou como o microcontrolador responsável pelo processamento das imagens, e o "3-Watt Speaker", no valor de 6,29 reais, utilizado para fornecer feedback sonoro ao usuário. O "MAX98357A I2S 3-Watt Class D Amplifier", que custou 7,39 reais, foi incorporado para melhorar a qualidade do som emitido pelo speaker, garantindo um áudio nítido e claro, essencial para interações com o sistema. Para fornecer energia ao sistema, o "5V USB Power

Supply Cable", com preço de 6,89 reais, foi escolhido como fonte de alimentação, garantindo que todos os componentes funcionassem de maneira estável. O custo do servidor não foi incluído no projeto, já que se trata de um sistema embarcado.

Em termos de viabilidade, o custo total dos componentes foi considerado acessível, somando ao todo R\$154,16. Além disso, o projeto de reconhecimento facial se mostrou funcional, com todos os componentes trabalhando de forma integrada para proporcionar uma experiência eficiente.

5 CONCLUSÃO

A realização deste estudo teve como objetivo principal desenvolver e avaliar um sistema de reconhecimento facial voltado para auxiliar pessoas com Alzheimer, considerando fatores adversos como variação de iluminação, desfoque na imagem e diferentes graus de tolerância no reconhecimento. A análise dos experimentos realizados permitiu compreender as limitações e potencialidades do sistema em diversas condições, garantindo uma visão abrangente sobre seu desempenho e confiabilidade.

Os testes de luminosidade evidenciaram que o sistema possui dependência da presença de luz para a identificação correta dos rostos. Em ambientes de extrema escuridão, a impossibilidade de capturar informações visuais resultou em falhas no reconhecimento, o que era esperado, dado que os algoritmos empregados dependem da extração de características faciais para efetuar a identificação. No entanto, ao introduzir fontes artificiais de luz, mesmo que de baixa intensidade, o sistema se mostrou eficiente, demonstrando que pode operar de maneira confiável em ambientes com iluminação moderada a intensa. Esses resultados ressaltam a importância de garantir condições mínimas de luminosidade para o uso do sistema no cotidiano, sugerindo que futuras versões poderiam incorporar tecnologias de visão noturna ou infravermelho para mitigar essa limitação.

Em relação ao teste de blur, foi constatado que o sistema conseguiu manter um bom desempenho até um grau de blur de 60. Entretanto, em níveis mais elevados, especificamente nos testes com blur 80 e 99, houve falhas no reconhecimento, indicando que a presença de um desfoque intenso compromete a extração de informações faciais essenciais. Esse resultado reforça a necessidade de manutenção periódica das câmeras utilizadas, evitando o acúmulo de sujeira ou umidade que possam degradar a qualidade das imagens capturadas e, conseqüentemente, reduzir a precisão do reconhecimento facial.

A análise dos dados permitiu identificar padrões relevantes no comportamento do sistema. Os tempos de resposta das medições realizadas apresentaram forte correlação entre si, indicando estabilidade na execução do reconhecimento facial ao longo dos testes. Além disso, verificou-se que a inclinação da face exerce um impacto moderado no tempo de reconhecimento e na taxa de acertos, com maiores dificuldades ocorrendo em ângulos mais acentuados, como 45° e 90°. Esse aspecto sugere que aprimoramentos nos algoritmos de detecção de pontos faciais podem contribuir para uma melhor adaptação a variações de posição do rosto, tornando o sistema mais robusto para diferentes cenários de uso.

Outro fator determinante para o sucesso do reconhecimento foi a tolerância do sistema. Os resultados mostraram que valores baixos de tolerância, entre 0,1 e 0,5, restringem excessivamente a margem de variação permitida para a identificação da face, levando a um aumento nas falhas de reconhecimento. Por outro lado, valores superiores a 0,6 começaram a apresentar taxas de acerto mais elevadas, com o melhor desempenho sendo registrado na tolerância de 0,7. Esse resultado destaca a necessidade de calibrar adequadamente esse parâmetro para equilibrar

precisão e flexibilidade no reconhecimento facial, permitindo que o sistema funcione de maneira eficaz mesmo diante de pequenas variações nas características faciais do usuário.

A relação entre desfoque e tempo de resposta também revelou um comportamento interessante. Inicialmente, à medida que o blur aumentava, houve um incremento no tempo de resposta do sistema, indicando que o algoritmo exigia um maior processamento para realizar a identificação em imagens parcialmente degradadas. No entanto, em níveis de desfoque extremos, o tempo de resposta caiu drasticamente, o que pode ser atribuído à ausência de detalhes suficientes para que o sistema iniciasse uma tentativa efetiva de reconhecimento. Esse comportamento reforça a importância de estabelecer um limiar adequado para a qualidade das imagens capturadas, evitando que o sistema opere com dados insuficientes para uma identificação confiável.

Dessa forma, os experimentos realizados permitiram validar a eficácia do sistema em condições controladas, ao mesmo tempo em que evidenciaram seus desafios e limitações. A dependência da iluminação, a sensibilidade ao desfoque elevado e a necessidade de ajustes na tolerância são fatores que devem ser considerados em desenvolvimentos futuros, visando aprimorar a robustez e a aplicabilidade do sistema em cenários reais.

5.1 Trabalhos Futuros

Para aprimorar o sistema de reconhecimento facial desenvolvido, futuras melhorias podem ser implementadas com foco na praticidade e viabilidade do sistema. Uma das possibilidades é a otimização do pré-processamento das imagens, aplicando ajustes automáticos de brilho e contraste para melhorar o desempenho do reconhecimento em condições adversas, como baixa iluminação e desfoque moderado. Além disso, a calibração dos parâmetros de tolerância do reconhecimento pode contribuir para um melhor equilíbrio entre precisão e taxa de erros.

Outra melhoria viável consiste na implementação de um mecanismo de atualização incremental do banco de dados de rostos. Essa funcionalidade possibilitaria a adição contínua de novas imagens de referência, aprimorando a confiabilidade do reconhecimento facial sem a necessidade de reprocessar todo o conjunto de dados.

A integração do sistema com dispositivos móveis também representa um avanço relevante e de fácil implementação. Um aplicativo simples para smartphones poderia ser desenvolvido para exibir notificações ou permitir a autorização remota de acessos, tornando o uso do reconhecimento facial mais intuitivo e acessível.

Além disso, uma abordagem complementar para aumentar a segurança do sistema seria a implementação de um segundo fator de autenticação, como a utilização de um código PIN ou QR Code. Essa estratégia reduziria o risco de falsos positivos e garantiria uma camada adicional de verificação em situações onde a confiabilidade do reconhecimento facial seja comprometida.

Por fim, a otimização do código-fonte e a utilização de estratégias de processamento paralelo podem reduzir o tempo de resposta do sistema, tornando-o mais eficiente para aplicações em dispositivos de hardware limitado, como o ESP32. A redução da resolução das imagens analisadas, desde que sem comprometer a precisão do reconhecimento, pode ser uma alternativa para melhorar o desempenho computacional e viabilizar a adoção do sistema em aplicações embarcadas de baixo custo.

Essas melhorias, além de contribuírem para a evolução do sistema, possibilitam sua aplicação em um número maior de cenários, garantindo maior eficiência e acessibilidade.

REFERÊNCIAS

- ALATTAR, M.; GOVIND, A.; MAINALI, S. Artificial intelligence models for the automation of standard diagnostics in sleep medicine—a systematic review. **Bioengineering**, v. 11, n. 3, p. 206, 2024. Citado na página 18.
- ALJOJO, N. *et al.* Alzheimer assistant: a mobile application using machine learning. **Romanian Journal of Information Technology and Automatic Control**, v. 30, n. 4, p. 7–26, 2020. Citado 3 vezes nas páginas 14, 22 e 23.
- BELLER, E. *et al.* Making progress with the automation of systematic reviews: principles of the international collaboration for the automation of systematic reviews (icasr). **Systematic Reviews**, v. 7, p. 1–7, 2018. Citado na página 14.
- BERNARDO, D. H. V.; SILVA, W. V. da; LOPES, A. M. Z. Dispositivo de automação de residência para pacientes com doença de alzheimer: Um protótipo em arduino. **Revista Fatecnológica da Fatec-Jahu**, v. 17, n. 1, p. 105–117, 2023. Citado na página 20.
- BRADSKI, G. The opencv library. **Dr. Dobb's Journal of Software Tools**, 2000. Citado na página 29.
- CHAN, J. J. H. *et al.* Facial artificial intelligence in ophthalmology and medicine: fundamental and transformative applications. **Therapeutic Advances in Ophthalmology**, v. 16, p. 25158414241302871, 2024. Citado na página 21.
- ELIF, K.; TAŞKAPILIOĞLU, ; BAKAR, M. Caregiver burden in different stages of alzheimer's disease. **Archives of Neuropsychiatry**, v. 54, n. 1, p. 82, 2017. Citado na página 14.
- FARDOUN, H. M.; MASHAT, A. A.; CASTILLO, J. R. Recognition of familiar people with a mobile cloud architecture for alzheimer patients. **Disability and Rehabilitation**, v. 39, n. 4, p. 398–402, 2017. Citado na página 21.
- JOSHI, V. M. *et al.* Demnet neurodeep: Alzheimer detection using electroencephalogram and deep learning. **Bulletin of Electrical Engineering and Informatics**, v. 14, n. 1, p. 457–465, 2025. Citado na página 20.
- KADHIM, T. A. *et al.* A face recognition application for alzheimer's patients using esp32-cam and raspberry pi. **Journal of Real-Time Image Processing**, v. 20, n. 5, p. 100, 2023. Citado 2 vezes nas páginas 21 e 22.
- LIU, H.; KADIR, A.; LIU, J. Keyed hash function using hyper chaotic system with time-varying parameters perturbation. **IEEE Access**, v. 7, p. 37211–37219, 2019. Citado na página 27.
- LODHA, P.; TALELE, A.; DEGAONKAR, K. Diagnosis of alzheimer's disease using machine learning. In: **2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)**. [S.l.]: IEEE, 2018. p. 1–4. Citado na página 20.
- LUGARESI, C. *et al.* Mediapipe: A framework for building perception pipelines. **arXiv preprint arXiv:1906.08172**, 2019. Citado na página 26.
- NAIK, G.; BHIDE, S. S. Will the future of knowledge work automation transform personalized medicine? **Applied & Translational Genomics**, v. 3, n. 3, p. 50–53, 2014. Citado na página 18.

OBE, O.; ABE, S.; BOYINBODE, O. Development of wireless home automation system for the disabled (deaf, dumb and alzheimer) people. **International Journal of Scientific & Engineering Research**, v. 9, n. 10, p. 1–6, 2018. Citado na página 19.

PRATIK, A.; ABHIMANYU, A.; PATIL, B.; DHUMAL, A. Optimized tolerance based student's face recognition and identification using deep learning. **Applied Computer Vision and Soft Computing with Interpretable AI**, Taylor & Francis Group, p. 8, 2023. Citado na página 42.

RAPARTHI, M. Robotic process automation in healthcare-streamlining precision medicine workflows with ai. **Journal of Science & Technology**, v. 1, n. 1, p. 91–99, 2020. Citado na página 18.

REZAEI, M. *et al.* **Emerging Technologies in Medicine: Artificial Intelligence, Robotics, and Medical Automation**. [S.l.]: Kindle, 2023. v. 3. 1-184 p. Citado na página 18.

ROOPAELI, M.; RAD, P.; PREVOST, J. J. A wearable iot with complex artificial perception embedding for alzheimer patients. In: **2018 World Automation Congress (WAC)**. [S.l.]: IEEE, 2018. p. 1–6. Citado na página 21.

SCHELTS, P. *et al.* Alzheimer's disease. **The Lancet**, v. 388, n. 10043, p. 505–517, 2016. Citado na página 14.

SERENIKI, A.; VITAL, M. A. B. F. A doença de alzheimer: aspectos fisiopatológicos e farmacológicos. **Revista de Psiquiatria do Rio Grande do Sul**, v. 30, 2008. Citado na página 14.

WANG, Q. *et al.* Blockchain for the iot and industrial iot: A review. **Internet of Things**, v. 10, p. 100081, 2020. Citado na página 18.

.1 Códigos

.1.1 Código de Cadastro

```
# Importação das bibliotecas
import numpy as np
import os
import cv2
import json
import face_recognition as fr

# Caminho correto para a pasta "fotos"
fotos_path = r"C:\TESTE\PPI1\fotos"
json_filename = "face_encodingsR1.json"

# Listas para armazenar nomes e codificações
reference_names = []
all_encodings = []

# Verificar se a pasta "fotos" existe
if os.path.exists(fotos_path) and os.path.isdir(fotos_path):
    for filename in os.listdir(fotos_path):
        if filename.lower().endswith(('.png', '.jpg', '.jpeg')):
            image_path = os.path.join(fotos_path, filename)
            image = cv2.imread(image_path)

            if image is None:
                print(f"Erro ao carregar a imagem: {filename}.
                Verifique se o arquivo é válido.")
                continue

            rgb_image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
            # Converter para RGB

            # Detectar e codificar rostos
            face_encodings = fr.face_encodings(rgb_image)
            if face_encodings:
                all_encodings.append(face_encodings[0].tolist())
                # Salvar como lista
                reference_names.append(os.path.splitext(filename)[0])
```

```
        else:
            print(f"Aviso: Nenhum rosto detectado em {filename},
                  ignorando.")

if all_encodings:
    # Dicionário para salvar os dados
    data_to_save = {
        "reference_names": reference_names,
        "all_encodings": all_encodings
    }

    # Salvar os dados no JSON
    with open(json_filename, "w") as json_file:
        json.dump(data_to_save, json_file)

    print(f"Dados salvos em {json_filename}")
else:
    print("Nenhuma codificação facial foi gerada. Verifique se
          há rostos nas imagens.")
else:
    print(f"A pasta '{fotos_path}' não foi encontrada!")
```

.1.2 Código de Reconhecimento Facial

```
import face_recognition as fr
import numpy as np
import os
import cv2
import json
import pyttsx3
import time

# Inicializar o motor de síntese de voz
engine = pyttsx3.init()

# Carregar as codificações e nomes de referência a partir do arquivo
JSON
with open(r"C:\TESTE\PPI1\face_encodingsR1.json", "r") as json_file:
```

```
data = json.load(json_file)
reference_names = data["reference_names"]
all_encodings = data["all_encodings"]

# Função para fazer o assistente falar
def speak(text):
    engine.say(text)
    engine.runAndWait()

def check_face(frame, reference_encodings, reference_names, blur_level,
last_recognized_time, last_recognized_name):
    faces_frame = fr.face_locations(frame)
    if len(faces_frame) > 0:
        encodings_frame = fr.face_encodings(frame, faces_frame)

        for encoding_frame, face_location in zip(encodings_frame,
faces_frame):
            distances = fr.face_distance(reference_encodings,
encoding_frame)
            tolerance_percentage = 0.7

            min_distance_index = np.argmin(distances)
            min_distance = distances[min_distance_index]

            if min_distance < tolerance_percentage*np.mean(distances):
                match_name = reference_names[min_distance_index]
                cv2.putText(frame, f"Reconhecido: {match_name}",
(20, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)

                # Verificar se passaram 5 segundos desde a última fala
                current_time = time.time()
                if current_time - last_recognized_time > 5 and
match_name != last_recognized_name:
                    speak(match_name)
                    print(match_name)
                    last_recognized_time = current_time
                    last_recognized_name = match_name

            else:
```

```
        cv2.putText(frame, "Nao reconhecido", (20, 50),
                    cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)

    # Desfoque a área da face
    (top, right, bottom, left) = face_location
    face_region = frame[top:bottom, left:right]

    if blur_level > 0:
        ksize = (blur_level * 2 + 1, blur_level * 2 + 1)
        face_region = cv2.GaussianBlur(face_region, ksize, 0)
        frame[top:bottom, left:right] = face_region

    return last_recognized_time, last_recognized_name

# Iniciar a webcam
#cap = cv2.VideoCapture('http://10.70.81.103:8080/video')
cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 640)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)

blur_level = 0 # Nível inicial de desfoque
last_recognized_time = 0 # Tempo da última fala
last_recognized_name = "" # Último nome reconhecido

while True:
    ret, frame = cap.read()
    if ret:
        last_recognized_time, last_recognized_name = check_face
            (frame, all_encodings, reference_names, blur_level,
             last_recognized_time, last_recognized_name)

        cv2.imshow('Webcam Face Recognition', frame)

    key = cv2.waitKey(1) & 0xFF
    if key == ord('q'):
        break
    elif key == ord('+'):
        blur_level = min(blur_level + 5, 99)
        print(f"Nível de desfoque: {blur_level}")
```

```
        # Imprime o nível de desfoque
    elif key == ord('-'):
        blur_level = max(blur_level - 1, 0)
        print(f"Nível de desfoque: {blur_level}")    # Imprime o nível
        de desfoque

cap.release()
cv2.destroyAllWindows()
```

.1.3 Código da Câmera

```
#include "esp_camera.h"
#include <WiFi.h>
#include "esp_timer.h"
#include "img_converters.h"
#include "Arduino.h"
#include "fb_gfx.h"
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "dl_lib.h"
#include "esp_http_server.h"

// Configuração da rede WiFi
const char* ssid = "nome-da-rede-wifi";
const char* password = "senha-da-rede-wifi";
#define PART_BOUNDARY "123456789000000000000987654321"

// Configuração do modelo de câmera (CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0
#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27
#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
```

```
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM 23
#define PCLK_GPIO_NUM 22

static const char* _STREAM_CONTENT_TYPE = "multipart/x-mixed-replace;
boundary=" PART_BOUNDARY;
static const char* _STREAM_BOUNDARY = "\r\n--" PART_BOUNDARY "\r\n";
static const char* _STREAM_PART = "Content-Type: image/jpeg\r\nContent
-Length: %u\r\n\r\n";

httpd_handle_t stream_httpd = NULL;

static esp_err_t stream_handler(httpd_req_t* req) {
    camera_fb_t* fb = NULL;
    esp_err_t res = ESP_OK;
    size_t _jpg_buf_len = 0;
    uint8_t* _jpg_buf = NULL;
    char* part_buf[64];

    res = httpd_resp_set_type(req, _STREAM_CONTENT_TYPE);
    if (res != ESP_OK) {
        return res;
    }

    while (true) {
        fb = esp_camera_fb_get();
        if (!fb) {
            Serial.println("Camera capture failed");
            res = ESP_FAIL;
        } else {
            if (fb->width > 400) {
                if (fb->format != PIXFORMAT_JPEG) {
                    bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf,
                    &_jpg_buf_len);
                    esp_camera_fb_return(fb);
                }
            }
        }
    }
}
```

```
        fb = NULL;
        if (!jpeg_converted) {
            Serial.println("JPEG compression failed");
            res = ESP_FAIL;
        }
    } else {
        _jpg_buf_len = fb->len;
        _jpg_buf = fb->buf;
    }
}

if (res == ESP_OK) {
    size_t hlen = snprintf((char*)part_buf, 64, _STREAM_PART,
        _jpg_buf_len);
    res = httpd_resp_send_chunk(req, (const char*)part_buf,
        hlen);
}

if (res == ESP_OK) {
    res = httpd_resp_send_chunk(req, (const char*)_jpg_buf,
        _jpg_buf_len);
}

if (res == ESP_OK) {
    res = httpd_resp_send_chunk(req, (const char*)
        _STREAM_BOUNDARY, strlen(_STREAM_BOUNDARY));
}

if (fb) {
    esp_camera_fb_return(fb);
    fb = NULL;
    _jpg_buf = NULL;
} else if (_jpg_buf) {
    free(_jpg_buf);
    _jpg_buf = NULL;
}

if (res != ESP_OK) {
```

```
        break;
    }
    // Serial.printf("MJPG: %uB\n", (uint32_t) (_jpg_buf_len));
}

return res;
}

void startCameraServer() {
    httpd_config_t config = HTTPD_DEFAULT_CONFIG();
    config.server_port = 80;

    httpd_uri_t index_uri = {
        .uri = "/",
        .method = HTTP_GET,
        .handler = stream_handler,
        .user_ctx = NULL
    };

    // Serial.printf("Starting web server on port: '%d'\n",
    config.server_port);
    if (httpd_start(&stream_httpd, &config) == ESP_OK) {
        httpd_register_uri_handler(stream_httpd, &index_uri);
    }
}

void setup() {
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
    Serial.begin(115200);
    Serial.setDebugOutput(false);

    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
```

```
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
config.frame_size = FRAMESIZE_UXGA;
config.jpeg_quality = 10;
config.fb_count = 2;

// Iniciação da câmera
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
}

// Conexão WiFi
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

// Início da transmissão no servidor Web
startCameraServer();
Serial.print("Camera Stream Ready! Go to: http://");
Serial.print(WiFi.localIP());
}
```

```
void loop() {  
    delay(1);  
}
```