

UNIVERSIDADE CESUMAR - UNICESUMAR
CENTRO DE CIÊNCIAS EXATAS TECNOLÓGICAS E AGRÁRIAS CURSO DE
GRADUAÇÃO ENG. CONTROLE E AUTOMAÇÃO

**PESQUISA DE VIABILIDADE E FUNCIONAMENTO DE SISTEMA DE EYE-
TRACKING PARA DISPOSITIVOS DE REALIDADE VIRTUAL (RV)**

Artigo apresentado ao Curso de Graduação em Engenharia De Controle e Automação da Universidade Cesumar – UNICESUMAR como requisito parcial para a obtenção do título de Bacharel(a) em Eng. Mecatrônica, sob a orientação do Prof. Dr. Fernando Pereira Calderaro.

MARINGÁ – PR
2024

FOLHA DE APROVAÇÃO

NOME DO ALUNO: PEDRO GARCIA PAGAN NETO

TÍTULO DO TRABALHO: PESQUISA DE VIABILIDADE E FUNCIONAMENTO DE SISTEMA DE EYE-TRACKING PARA DISPOSITIVOS DE REALIDADE VIRTUAL (RV)

Artigo apresentado ao Curso de Graduação em engenharia de controle e automação da Universidade Cesumar UNICESUMAR como requisito parcial para a obtenção do título de Bacharel(a) em engenharia de controle e automação, sob a orientação do Prof. Dr. Fernando Pereira Calderaro.

Aprovado em: ____ de _____ de ____.

BANCA EXAMINADORA

Nome do professor – (Titulação, nome e Instituição)

Nome do professor - (Titulação, nome e Instituição)

Nome do professor - (Titulação, nome e Instituição)

RESUMO:

O presente estudo investiga a viabilidade e o funcionamento de um sistema de eye-tracking para dispositivos de realidade virtual. O objetivo principal é desenvolver um protótipo de baixo custo que permita a implementação dessa tecnologia em dispositivos de RV que não possuem suporte nativo. A metodologia abrange a definição das peças necessárias, a programação de placas microcontroladoras, a montagem de hardware e a execução de testes. O protótipo utiliza emissores de luz infravermelha e câmeras para captar os movimentos oculares, proporcionando uma experiência mais imersiva ao usuário. Os resultados indicam a possibilidade de integrar a tecnologia em dispositivos de gerações anteriores, ampliando o acesso à funcionalidade de eye-tracking.

PALAVRAS-CHAVE: realidade virtual; rastreamento ocular; protótipo; imersão; tecnologia assistiva.

ABSTRACT:

This study investigates the feasibility and functionality of an eye-tracking system for virtual reality (VR) devices. The main objective is to develop a low-cost prototype that enables the implementation of this technology in VR devices that lack native support. The methodology includes defining the necessary components, programming microcontroller boards, hardware assembly, and conducting tests. The prototype uses infrared light emitters and cameras to capture eye movements, providing a more immersive experience for the user. The results indicate the possibility of integrating the technology into older-generation devices, expanding access to eye-tracking functionality.

KEYWORDS: virtual reality; eye-tracking; prototype; immersion; assistive technology.

1 Introdução

A realidade virtual é uma tecnologia que permite aos usuários imergirem em ambientes tridimensionais simulados, proporcionando experiências sensoriais que replicam ou divergem da realidade física. Esta imersão é alcançada através de dispositivos específicos, como óculos de realidade virtual, fones de ouvido, luvas sensoriais, entre outros, que interagem com *softwares* especialmente desenvolvidos para criar mundos virtuais.

Conforme a definição de Sherman e Craig (2003), realidade virtual é "um meio no qual o usuário, ao interagir com um ambiente tridimensional gerado por computador, tem a sensação de estar presente dentro desse ambiente". Essa tecnologia tem se expandido para diversas áreas, incluindo entretenimento, educação, medicina e treinamento militar, devido à sua capacidade de criar cenários realistas e controláveis (BURDEA; COIFFET, 2003).

Além disso, a RV tem se mostrado uma ferramenta eficaz em diversas aplicações, como no campo da educação, onde simulações interativas podem melhorar o aprendizado e a retenção de conhecimento dos alunos (FREINA; OTT, 2015).

Portanto, a realidade virtual não apenas oferece novas formas de entretenimento e educação, mas também abre possibilidades significativas para avanços em áreas científicas e práticas cotidianas, evidenciando seu impacto e potencial transformador na sociedade moderna.

Dentro do mercado de realidade virtual, podemos encontrar diversos módulos tecnológicos que implementam a experiência e imersão do usuário, um claro exemplo disso se trata dos *eye-trackers* que, de acordo com Barreto (2012), tratam-se de “um conjunto de tecnologias que permite medir e registrar os movimentos oculares de um indivíduo perante a amostragem de um estímulo em ambiente real ou controlado”. Os *eye-trackers*, de acordo com Clay (2019), foram utilizados pela primeira vez durante o começo do século 20, utilizando uma lente de contato com ponteiro para monitorar o comportamento ocular.

Nos dias de hoje, devido aos avanços tecnológicos na área, a implementação mais comum dos *eye-trackers* se trata da utilização de emissão de luz infravermelha e a captação dessa por uma câmera, esse processo, de acordo com o site Tobii Dynavox (2024, acessado em; 12/05/2024), funciona de forma que o dispositivo de *eye-tracking* emite luzes infravermelhas através dos dispositivos de emissão presentes na tecnologia, essas luzes então são refletidas pelo olho do usuário e então captadas por uma câmera conectada ao dispositivo, que irá então fazer a tradução das luzes recebidas em dados.

No contexto de RV, a presença de um dispositivo de *eye-tracking* permite uma melhor calibração para o visor de RV, além de também possibilitar o envio dos dados captados pelo *eye-tracking* para *softwares* externos e/ou jogos, como é o caso do *VRChat*, um jogo que permite ao usuário interagir socialmente com pessoas de qualquer lugar do mundo, tendo a utilização de dispositivos de RV como o foco principal para tornar essas interações mais imersivas e mais interessantes para o usuário.

Ao que diz respeito aos dispositivos de RV, muitos dos dispositivos mais recentes são desenvolvidos com sistemas de *eye-tracking* já instalados, como é o caso do Quest Pro e do VIVE Pro Eye, ou possuem módulos que podem ser comprados a parte para permitir tal tecnologia no dispositivo, como é o caso do dispositivo VIVE XR Elite, possuindo o módulo VIVE Full Face Tracker para permitir a funcionalidade de *eye-tracking* além de também um *tracking* facial.

Porém, da mesma forma que dispositivos mais recentes possuem a tecnologia de forma muito mais acessível, dispositivos de modelos mais antigos não possuem uma viabilidade para tal tecnologia de forma oficial, como é o caso do Valve Index, outros modelos do Quest e muitos outros dispositivos que não possuem *eye-tracking* já instalado e nem módulos para acessar tal tecnologia.

No contexto brasileiro, dispositivos de realidade virtual são considerados um luxo, uma vez que se trata de uma tecnologia pouco disponível no território e também possuem um preço elevado para a população, isso faz com que versões dos dispositivos e módulos com *eye-tracking* sejam economicamente inviáveis, visto que os mesmos são ainda mais recentes e, portanto, mais custosos.

Diante do contexto apresentado acima, o projeto se trata de uma Pesquisa Aplicada exploratória e mista, visando estabelecer a viabilidade e o funcionamento de um protótipo de *eye-tracking* desenvolvido pela ETVR, licenciado sob a *GNU General Public License* versão 3 (GPL-3.0) (EyeTrackVR, 2024). O protótipo permite, em tese, que dispositivos de *eye-tracking* para óculos de RV sejam desenvolvidos em casa, diminuindo custos com módulos para aqueles que possuem um dispositivo compatível e permitindo que dispositivos de linhas mais ultrapassadas tenham acesso à tecnologia.

Na etapa de metodologia, serão definidas as etapas pelo qual o trabalho irá se situar, definindo a ordem dos processos que serão executados.

2 Metodologia

Levando em conta o contexto acima, o desenvolvimento desse projeto trará um melhor entendimento de sistemas de *eye-tracking*, checando a possibilidade da implementação e teste do mesmo em óculos de RV que não possuem o mesmo já instalados. Tal pesquisa permitirá, se conclusiva, confirmar a funcionalidade de um sistema de *eye-tracking DIY (do it yourself)* e levar ao usuário uma experiência mais imersiva quando utilizando o dispositivo de RV.

Para a implementação dessa tecnologia o projeto contará com algumas etapas para a sua montagem, sendo essas: etapa de definição das peças; etapa de programação de possíveis placas/Arduino; etapa de montagem; etapa de instalação de *software* e, por fim, etapa de teste.

2.1 Definição de peças

2.1.1 Emissor de *IR* (luz infravermelha)

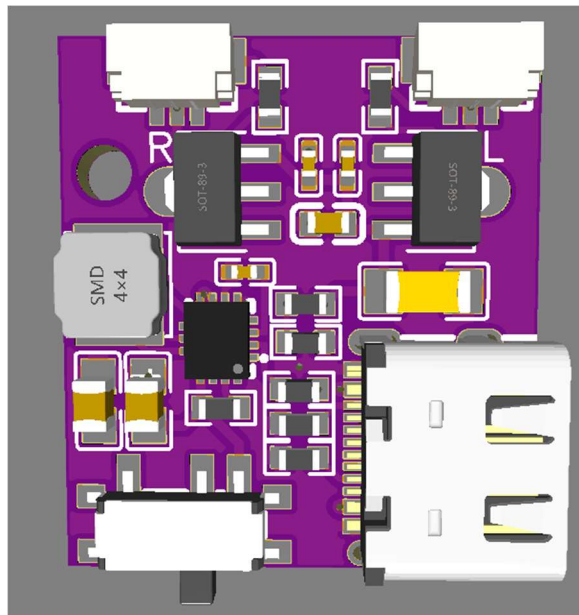
Como dito anteriormente, sistemas de *eye-tracking* nos dias de hoje atuam em um sistema que possua um agente emissor de luz infravermelha, a mesma é uma forma de radiação eletromagnética com comprimentos de onda superiores aos da luz visível, variando de aproximadamente 700 nanômetros (nm) a 1 milímetro (mm). Ela é classificada em três categorias principais: IR-A (700 nm a 1.400 nm), IR-B (1.400 nm a 3.000 nm) e IR-C (3.000 nm a 1 mm).

Para a elaboração do protótipo, deve-se levar em consideração a segurança do usuário em relação à exposição à luz infravermelha, e essa depende da intensidade e do tempo de exposição. Segundo a Comissão Internacional de Proteção contra Radiações Não-Ionizantes (ICNIRP), para exposições ocupacionais, o limite de exposição para o infravermelho próximo (IR-A) é de 100 W/m² para um período de 8 horas diárias (ICNIRP, 2013). Para o infravermelho médio (IR-B) e distante (IR-C), os limites são mais restritivos devido ao maior potencial de causar danos térmicos aos tecidos humanos, especialmente aos olhos e à pele.

Além do citado acima, de acordo com a *American Conference of Governmental Industrial Hygienists* (ACGIH), a exposição ao infravermelho não deve exceder 10 mW/cm² para períodos curtos (ACGIH, 2001). Essas diretrizes visam prevenir efeitos adversos como queimaduras e lesões oculares, incluindo a catarata induzida pelo calor.

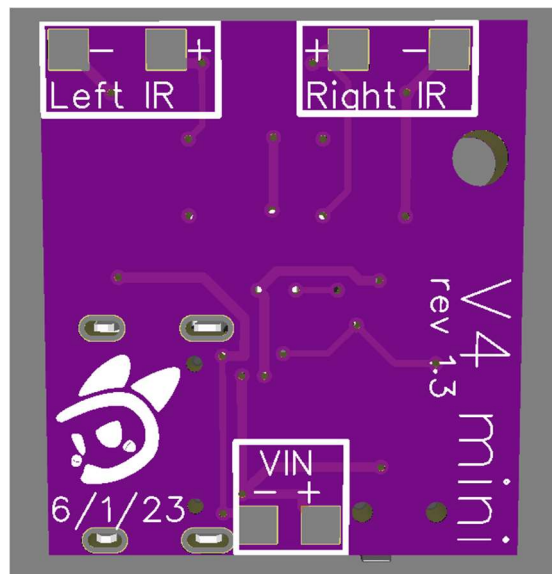
Portanto, para a definição do emissor de infravermelho, será utilizado um sistema de emissão de *IR* que fique abaixo dos valores citados acima, trazendo uma experiência segura ao usuário da tecnologia. A peça escolhida para o protótipo foi então o *V4 mini Fully Solder-less LED Kit*, desenvolvido pela ETVR, a peça possui 1 placa de controle alimentada por uma conexão USB-C que energiza 8 LEDs de *IR*, sendo 4 para cada um dos olhos. Seguem as figuras 1 e 2 para referência do microcontrolador:

Figura 1 - Placa *V4 mini Fully Solder-less LED Kit* (visão superior)



Fonte: Store, *EYETRACKVR*, 2024

Figura 2 - Placa *V4 mini Fully Solder-less LED Kit* (visão inferior)



Fonte: Store, *EYETRACKVR*, 2024

O hardware citado acima tem uma limitação de aproximadamente 2.4mA em sua corrente para garantir uma emissão de 0.6mW/cm² por LED, multiplicando esse valor por 4 teremos um total de 2.4mW/cm² por olho, se adequando aos limites de exposição considerados seguros ao ser humano.

2.1.2 Sistema para captura de imagens

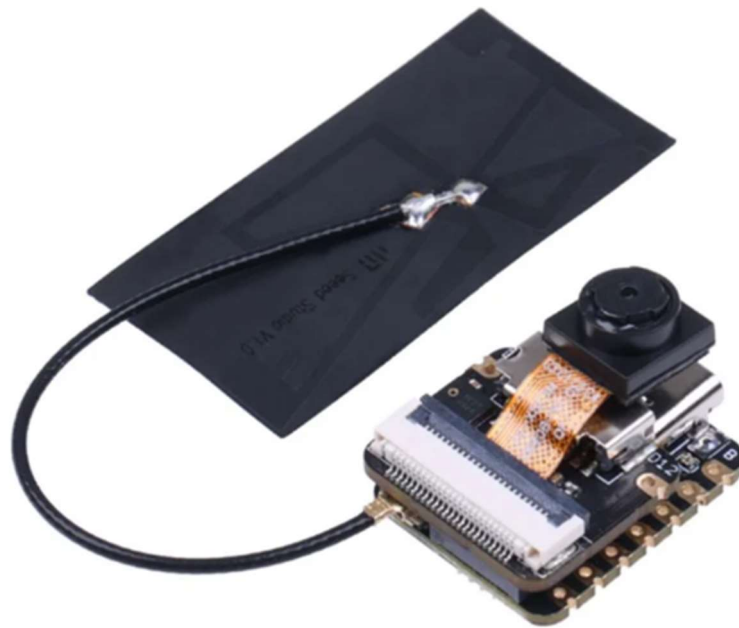
Para a construção do protótipo também se fará necessário a utilização de um sistema de câmeras para a captura da reflexão da luz *IR* no olho do usuário, para isso deve-se levar em consideração o tamanho do espaço vago presente na parte interior do dispositivo de RV. Para a grande maioria dos óculos de RV presentes no mercado o espaço vago presente ao redor das lentes é bem limitado e, no caso do protótipo, como esse será construído com o RV Valve Index em mente, esse espaço ficará ainda mais limitado.

Então, considerando os fatores acima, o sistema de câmeras que deverá ser utilizado será baseado em placas de arduino com câmeras, esses são dispositivos que combinam a plataforma de microcontroladores arduino com módulos de câmeras para capturar e processar imagens. Uma das principais vantagens dessas placas é a flexibilidade que oferecem para projetos de visão computacional. Por exemplo, elas podem ser utilizadas em projetos de reconhecimento de imagens, onde a câmera captura uma imagem e o Arduino processa os dados para identificar objetos ou padrões específicos. Esse tipo de aplicação é crucial em áreas como segurança, onde o reconhecimento de intrusos pode ser implementado de maneira eficiente e acessível (Craft, 2013).

Além disso, a simplicidade do Arduino facilita o desenvolvimento de protótipos rápidos e a experimentação com diferentes configurações de câmeras e algoritmos de processamento de imagens. Isso é particularmente útil em ambientes educacionais, onde estudantes podem aprender os princípios básicos da visão computacional e desenvolver suas próprias aplicações práticas (Packt, 2020).

Portanto, tendo-se definido que o protótipo utilizará de câmeras de arduino e que o espaço dentro do dispositivo de RV é limitado, foi-se escolhido 2 placas de arduino de modelo Seeed Studio XIAO ESP32-S3, juntamente com 2 câmeras OV2640 de 160° *night vision*, permitindo a detecção das luzes de *IR* e também possuam maior campo de visão para a captação de imagens. Segue as figuras 2 e 3 para referência dos materiais:

Figura 3 – Arduino Sseed Studio XIAO ESP32-S3



Fonte: AliExpress, 2024

Figura 4 – Câmera OV2640 160 degrees night vision

200MM
160 degrees
850nm
night vision



Fonte: AliExpress, 2024

Cada placa Seeed Studio XIAO ESP32-S3 será ligada em um dos módulos de câmera OV260 que estarão posicionados em cada um dos olhos, permitindo a captura de ambos os olhos de forma simultânea.

2.1.3 Fonte de alimentação e transmissão de dados

Considerando as peças acima, se fará necessário também uma fonte de alimentação de 5V para as placas de arduino e placa de *IR*. Tendo-se esse parâmetro em mente, foi-se escolhido a alimentação por portas USB 2.0, as quais também trabalham em 5V e permitem a transmissão de dados entre placa e dispositivo alimentador.

Considerando a quantidade de dispositivos que serão alimentados, a escolha para essa função será de um USB *hub*, os quais são dispositivos que expandem uma única porta USB em várias portas, permitindo a conexão de múltiplos dispositivos USB a uma única entrada. Esse *hub* será essencial para a alimentação do protótipo uma vez que o dispositivo de RV Valve Index possui apenas uma entrada USB em sua estrutura.

Um USB *hub* é um dispositivo que permite a expansão de uma única porta USB em várias, facilitando a conexão simultânea de diversos dispositivos a um computador. Este dispositivo é essencial para aumentar a capacidade de conectividade de computadores, especialmente em ambientes onde múltiplos periféricos são utilizados.

A importância das portas USB na tecnologia é notável, pois elas são fundamentais para a conexão e comunicação de periféricos com o computador. Além de permitir a transferência rápida de dados, essas portas fornecem energia, o que facilita a integração de diversos dispositivos em um único sistema. De acordo com Axelson (2015), os USB *hubs* são fundamentais para aumentar a quantidade de dispositivos que podem ser conectados a um único host, proporcionando maior flexibilidade e capacidade de expansão.

Tendo a alimentação definida, também se fará o uso de cabos USB 2.0 para USB tipo-C, permitindo que as placas de arduino sejam alimentadas a partir do USB *hub*.

2.1.4 Suporte para os componentes

Com a definição dos componentes de funcionamento, deve-se também elaborar um suporte para o acoplamento das peças no sistema de RV de forma a minimizar o impacto no conforto do usuário quando em uso.

Para a definição do material que será utilizado deve-se levar em conta o aquecimento das placas de arduino durante o funcionamento além de também desejar-se um material que seja minimamente resistente a impactos e torção, diminuindo as chances de quebra do mesmo durante a montagem do protótipo.

Com os fatores acima em mente, foi-se definido que o suporte das peças será feito a partir de uma impressora 3d. Uma impressora 3D é um dispositivo que cria objetos tridimensionais a partir de modelos digitais, geralmente adicionando camadas sucessivas de material, como plástico, resina ou metal. Este processo é conhecido como manufatura aditiva, em contraste com a manufatura subtrativa, que envolve a remoção de material. A tecnologia de impressão 3D é amplamente utilizada em diversas áreas, incluindo medicina, engenharia, arquitetura e design, permitindo a prototipagem rápida e a produção personalizada de peças complexas (GIBSON, 2015).

Considerando-se os fatores acima onde o material deve ser resistente a impactos e também possuir uma resistência térmica elevada, o material definido para o suporte foi o PETG (Polietileno Tereftalato Glicol). O material PETG (Polietileno Tereftalato Glicol) é amplamente utilizado na impressão 3D devido à sua combinação de durabilidade e facilidade de uso. Ele é conhecido por ser menos quebradiço que o PLA e mais fácil de imprimir do que o ABS, além de ser resistente a impactos e possuir boa estabilidade térmica. As temperaturas ideais para a impressão com PETG variam entre 220°C e 250°C para a extrusora e entre 70°C e 80°C para a mesa de impressão, a fim de garantir uma boa adesão durante o processo de impressão (CHUA e LEONG, 2015).

Lembrando-se que existem diversos modelos para o suporte e esses variam de diversas formas, deve-se escolher um modelo que atenda às necessidades citadas anteriormente de conforto, juntamente com a capacidade de ser instalado no modelo de RV especificado.

Quando se discute em relação em qual será a estrutura impressa, se utilizará o modelo encontrado no projeto Phys-Index-EyetrackVR-HW, pelo usuário Physics-Dude na plataforma GitHub, esse modelo permite a instalação do USB *hub* juntamente com as placas de arduino e

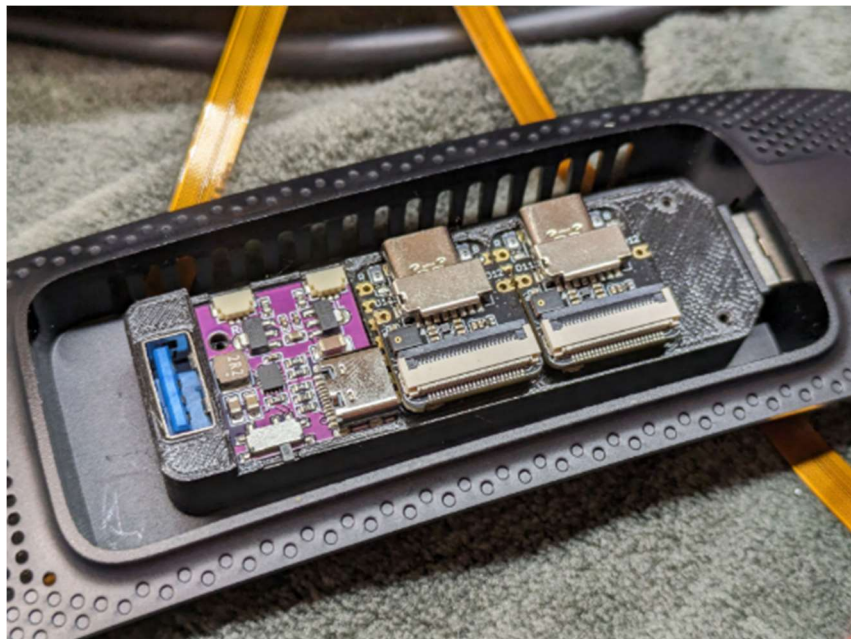
IR no mesmo compartimento, economizando uma grande quantidade de espaço além de também não alterar muito a estrutura original do dispositivo de RV, segue-se as figuras 1 e 2 para referência.

Figura 5 – Modelo Phys-Index-EyetrackVR-HW montado na placa frontal do dispositivo de RV



Fonte: Physics-Dude, GitHub, 2024

Figura 6 – Modelo Phys-Index-EyetrackVR-HW montado sem a placa de suporte frontal



Fonte: Physics-Dude, GitHub, 2024

2.2 Upload de Firmware

Para o funcionamento do protótipo, se faz de extrema importância que o *firmware* correto seja instalado nas placas de Arduino. O *firmware* de Arduino é um *software* gravado na memória de um microcontrolador da placa Arduino, responsável por controlar suas operações e interações com sensores, atuadores e outros componentes. Ele é essencial para definir o comportamento do hardware, permitindo a execução dos códigos de programação desenvolvidos pelo usuário (Massimo, 2022).

O *firmware* desejado para a execução e montagem do protótipo será instalado a partir do instalador de *firmwares* disponibilizado pela ETVR. Para se realizar o *flash* (atualização) do *firmware*, o usuário deverá conectar o microcontrolador Sseed Studio XIAO ESP32-S3 no computador através de um cabo de alimentação e passagem de dados enquanto segura o botão de *boot* presente no controlador, habilitando assim o modo *bootloader*.

Com o modo *bootloader* habilitado, o *software* irá solicitar qual o microcontrolador utilizado e, após selecionado, irá atualizar o *firmware* do microcontrolador para a versão necessária, permitindo que o mesmo funcione da forma desejada para o funcionamento do protótipo.

2.3 Montagem do protótipo

Mantendo-se em mente que os instrumentos que serão utilizados não foram citados na lista de materiais e tendo todas as peças principais listadas acima, pode-se dar início a montagem do protótipo.

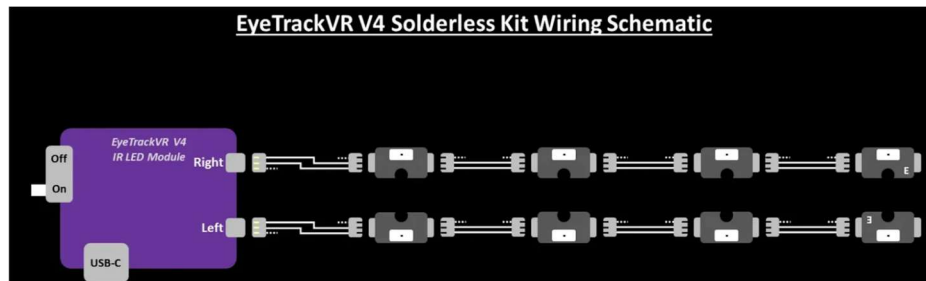
Inicialmente, para se ter uma base de montagem para o projeto, será feita a impressão 3d do modelo de suporte citado anteriormente em material PETG, esse processo poderá ser feito tanto no ambiente da universidade ou em outro ambiente que possua uma impressora 3d a disposição.

2.3.1 Montagem da estrutura externa do protótipo

Com a estrutura de suporte impressa, a preparação da placa de *IR* deve ser feita de acordo com as instruções de montagem da distribuidora (ETVR). As instruções ilustradas estão

disponíveis na figura 7 abaixo, onde se mostra como cada fio com os emissores de *IR* devem ser conectados na placa de controle:

Figura 7 – Instruções de montagem do *V4 mini*



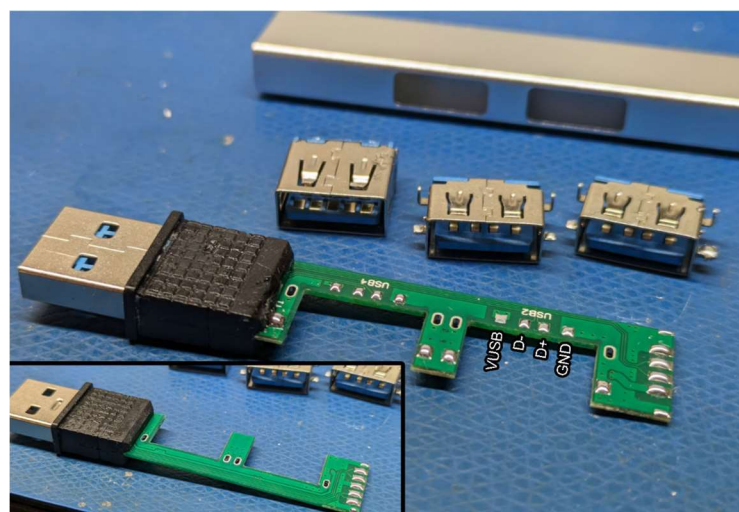
Fonte: Store, *EYETRACKVR*, 2024

No que se diz respeito as placas de Arduino, essas deverão ter seu *firmware* atualizado através das instruções citadas no tópico 2.2.1 com a utilização do *software* disponibilizado pela ETVR.

Com a estrutura impressa em mãos, será feito a remoção da estrutura externa do *hub* USB, garantindo que se sobre apenas a parte do circuito e a entrada de alimentação do mesmo. Esse processo deve ser feito cuidadosamente para não se danificar os circuitos internos do *hub*.

Serão soldados então alguns fios nos slots de alimentação presentes para cada um dos USBs do *hub*, mantendo-se em mente em qual slot cada fio foi soldado para uma instrução futura. Os slots de energia presentes onde se fica o USB em um USB *hub*, juntamente com o exemplo de como deve ficar a placa do *hub* estão expostos abaixo na figura 8:

Figura 8 – USB *hub* com entradas removidas e definições



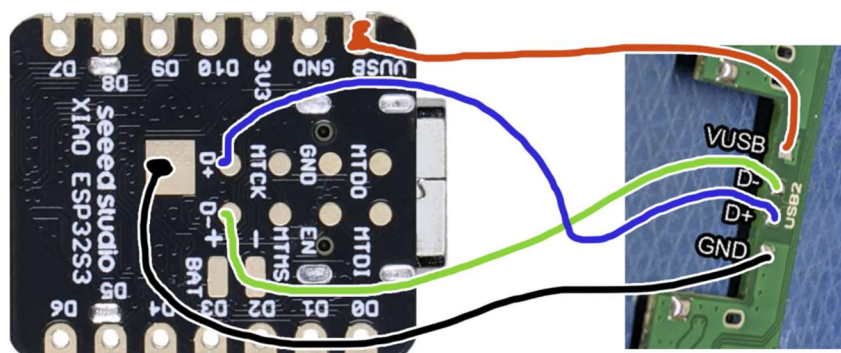
Fonte: Physics-Dude, GitHub, 2024

A placa interna do *hub* então será posicionada dentro do encaixe presente na estrutura impressa em PETG juntamente com uma das entradas USB que foram removidas no processo.

Com a placa interna posicionada, os fios soldados deverão ser passados pelo espaço interno presente na estrutura e, após isso, será também encaixado as duas placas Seeed Studio XIAO ESP32 na estrutura de PETG (tendo-se em mente que o suporte citado possui um espaço predefinido para esse encaixe).

Após o encaixe serão soldados os fios de cada entrada USB na sua devida conexão presente nas placas de Arduino para alimentá-las. A soldagem deve ser feita de acordo com o diagrama presente na figura 9 a seguir:

Figura 9 – Diagrama para alimentação das placas de Arduino



Fonte: Physics-Dude, GitHub, 2024

A partir das placas de Arduino também serão soldados os fios de alimentação da placa V4 (placa responsável pelo IR), garantindo que a mesma esteja sendo alimentada e funcionando corretamente.

Com o *hub* customizado montado, iremos conectar os dois módulos de câmera OV2640 nas placas de Arduino ESP32, tendo-se assim o funcionamento da parte responsável pela captura de imagens do protótipo.

Por fim, com toda essa estrutura preparada, basta-se conectar a saída de alimentação USB na entrada presente no dispositivo para a alimentação de todas as placas presente na estrutura.

2.3.2 Montagem da estrutura interna do protótipo

No que se diz respeito a montagem da parte das lentes, deve-se iniciar através das passagens dos fios necessários para o funcionamento da parte interna, ou seja, os fios de passagem de dados das câmeras OV2640 e também os fios que contém os *IRs* da placa V4.

Para a passagem desses fios, será removida a estrutura frontal do dispositivo de RV momentaneamente e esses fios serão passados através de um pequeno vão presente na parte superior dos circuitos, permitindo que os mesmos cheguem nas lentes do mesmo.

Com ambas as câmeras e os fios de *IR* presentes no interior da estrutura e próximos as lentes, utilizaremos a outra estrutura impressa em PETG e encaixaremos cada componente em seu devido lugar. A estrutura possui 1 *slot* de encaixe para câmera e 4 outros *slots* para cada emissor de *IR*, isso para cada lente presente no dispositivo de RV.

Com o posicionamento de cada componente na estrutura impressa, poderá ser feito o encaixe dessa estrutura diretamente ao redor de cada uma das lentes do dispositivo de RV, obtendo-se uma estrutura similar à exposta na figura 10.

Figura 10 – Estrutura montada ao redor da lente



Fonte: Physics-Dude, GitHub, 2024

2.4 Utilização do *software* e testes

Para se utilizar o protótipo montado, se conectará o dispositivo de RV no computador e deve-se iniciar o *software EyeTrackVR*, licenciado sob a *GNU General Public License* versão 3 (*GPL-3.0*) (*EyeTrackVR*, 2024).

Com o *software* em funcionamento, será possível a visualização de ambos os olhos que estão sendo capturados pelas câmeras internas do protótipo. A partir disso, o usuário deverá calibrar a posição de cada olho através do botão de calibração presente no *software* e então escolher um algoritmo para a tradução da imagem obtida pela câmera em dados.

O processo acima pode variar devido a como cada algoritmo se comporta em relação a diferentes imagens capturadas pelas câmeras do protótipo, logo deve-se testar cada um dos algoritmos presentes no *software* para verificar qual tem o desempenho que gera maior satisfação ao usuário.

Com o *software* em execução, o usuário agora deve abrir algum aplicativo que tem integração com sistemas de *eye-tracking*, no caso desse projeto, o aplicativo que será utilizado para o teste de funcionamento será o jogo de realidade virtual *VRChat*.

Dentro do jogo o usuário deve habilitar a opção de OSC, permitindo que o *software* se comunique com o jogo e envie os dados referentes à localização de cada uma das pupilas. A partir de então, os olhos do personagem dentro do aplicativo devem estar sincronizados com o do usuário, aumentando a imersão do mesmo dentro da plataforma.

3 Desenvolvimento

Para a realização do desenvolvimento do estudo, será construído um protótipo utilizando os passos e componentes citados acima, visando primariamente a análise de funcionamento e viabilidade do mesmo.

Inicialmente o projeto contaria com a estrutura presente nas figuras 5 e 6, porém devido a um problema em relação à soldagem do *hub* USB, foi-se necessária uma alteração, visando colocar o funcionamento do protótipo como prioridade.

3.1 Montagem

Iniciando-se a montagem, faz-se o desencaixe da placa frontal presente no dispositivo de RV Valve Index para que seja possível o acesso de sua parte interna, esse desencaixe é demonstrado na Figura 11 na página seguinte. Além do desencaixe da placa frontal, foi-se também desencaixada a almofada facial do dispositivo, abrindo espaço para os procedimentos ao redor das lentes.

Figura 11 – Desencaixe da placa frontal do dispositivo



Fonte: Autor

Após o desencaixe da placa frontal, também se foi necessário fazer um pequeno buraco, com a utilização de uma ferramenta aquecida para o corte, na parte superior da placa para facilitar a passagem de fios, como demonstrado na Figura 12.

Figura 12 – Buraco feito na placa frontal do dispositivo



Fonte: Autor

Continuando no processo de montagem, faz-se a preparação da estrutura que ficará ao redor da lente com as luzes de *IR* e a câmera para a captação das imagens.

Para esse processo foi-se impressa em material PETG a mesma estrutura demonstrada na Figura 10, utilizando a mesma como base. Então, após a impressão, foi feito o encaixe das placas de *IR* e câmeras OV2640 na estrutura (Figura 13) com a ajuda de uma pequena quantidade de cola quente, visando garantir a estabilidade das mesmas, uma vez que a impressão 3D da estrutura teve alguns pequenos erros e rebarbas em sua impressão.

Figura 13 – Estrutura montada ao redor das lentes



Fonte: Autor

Após essa montagem, os fios de ligação de ambas as câmeras e luzes *IR* foram passadas por um pequeno vão presente na parte superior interna do dispositivo (Figura 14, página seguinte), permitindo que esses fios sejam levados para o outro lado do dispositivo. Além disso, foi-se também encaixado novamente a almofada facial presente no dispositivo, permitindo assim que o mesmo seja utilizado.

Figura 14 – Vão presente na parte interna do dispositivo



Fonte: Autor

Com os fios presentes do outro lado, esses foram passados por dentro do pequeno buraco feito na placa frontal (Figura 11) e então a mesma foi encaixada novamente no dispositivo de RV (Figura 15).

Figura 15 – Placa frontal encaixada com passagem dos fios feita



Fonte: Autor

Seguindo para a próxima parte da montagem, foi anexada a placa *V4 mini Fully Solder-less LED Kit* e ambas as 2 placas Seeed Studio XIAO ESP32-S3 no dispositivo de RV com a ajuda de fita dupla face, como demonstrado na Figura 16.

Figura 16 – Fixação das placas controladoras no dispositivo



Fonte: Autor

Com os fios de conexão das câmeras e dos emissores de *IR* passados para o lado de fora, faz-se a conexão desses com suas respectivas placas de controle, sendo: os fios dos emissores conectados na placa do *V4 mini Fully Solder-less LED Kit* e os conectores das câmeras ligados em ambas as 2 placas Seeed Studio XIAO ESP32-S3, como demonstrado na Figura 17 (página seguinte).

Figura 17 – Fios conectados em suas determinadas placas



Fonte: Autor

Seguindo para a parte da alimentação das placas, conecta-se um USB *hub* na entrada de alimentação USB presente no dispositivo de RV e prende-se o *hub* na carcaça do dispositivo, também utilizando-se a fita dupla face no processo (Figura 18, página seguinte).

Figura 18 – USB *hub* anexado na carcaça do dispositivo



Fonte: Autor

Após a fonte de alimentação e passagem de dados ser anexada no dispositivo, resta apenas conectar essa às placas controladoras, utilizando 3 cabos USB tipo C para a conexão, como demonstrado na Figura 19.

Figura 19 – Conexão feita entre as placas controladoras e o USB *hub*



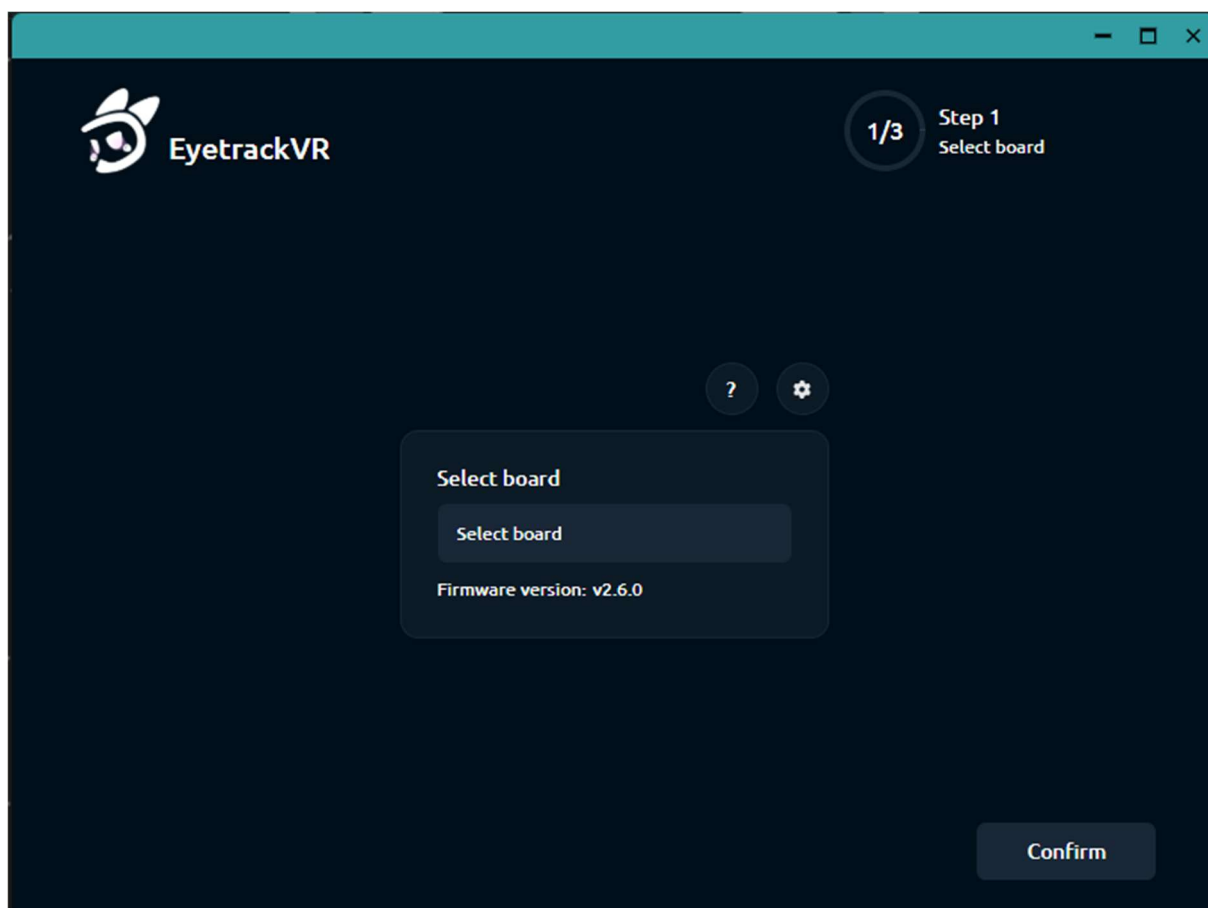
Fonte: Autor

Com esses processos finalizados, pode-se ver o protótipo estruturalmente pronto, necessitando agora seguir para a parte da programação das placas e do *software*.

3.2 Upload de *Firmware* nas placas controladoras

Seguindo para o próximo passo da prototipagem, tem-se a necessidade de atualizar o *firmware* presente nas placas Seeed Studio XIAO ESP32-S3. Para esse processo, foi-se feito o *download* do *software Firmware Flashing Tool* (Figura 20), distribuído gratuitamente pela ETVR e licenciado sob a *GNU General Public License* versão 3 (*GPL-3.0*) (EyeTrackVR, 2024).

Figura 20 – *Firmware Flashing Tool*



Fonte: Autor

Com o *software* instalado e em execução, coloca-se a placa que deseja atualizar o *firmware* em modo *bootloader*, para isso, é necessário pressionar o botão B (Figura 21) presente na placa que será atualizada e então conectá-la ao computador enquanto o botão se mantém

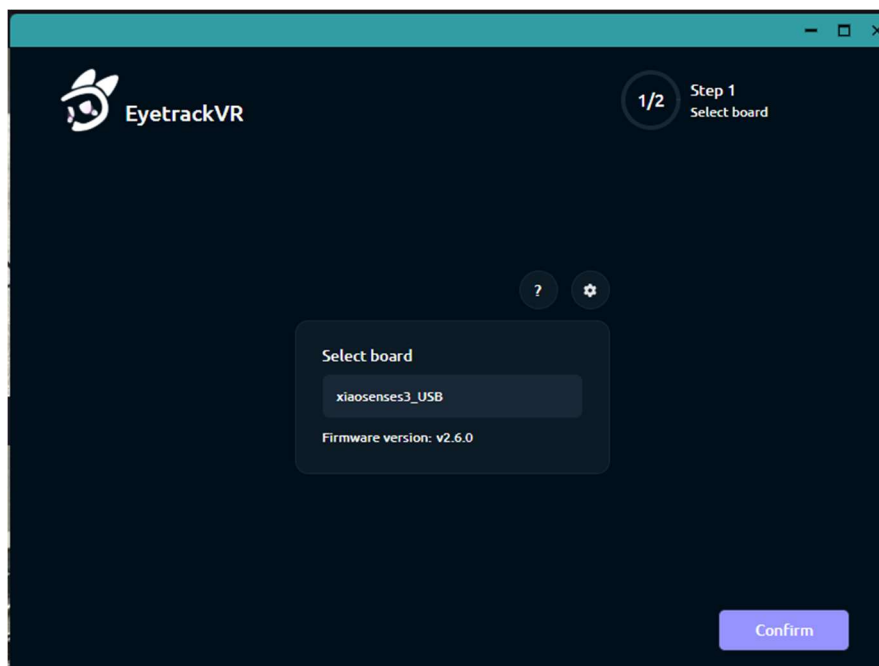
apertado. É importante que esse processo seja feito em uma placa de cada vez para não existirem conflitos nas portas USB do computador.

Figura 21 – Botão B presente na placa microcontroladora



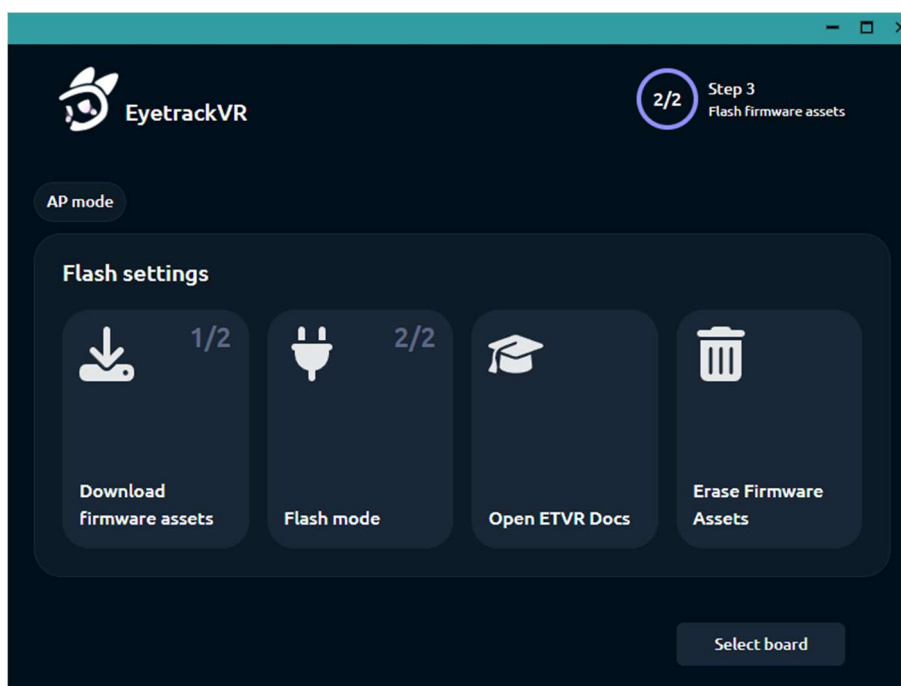
Fonte: Autor

Após em modo *bootloader*, faz-se a instalação do novo *firmware* na placa microcontroladora utilizando o *software Firmware Flashing Tool*. O processo será feito duas vezes, um para a cada placa SaaS Studio XIAO ESP32-S3. No *software* será selecionada a placa que deseja ser atualizada (Figura 22, página seguinte) e então o botão de confirmação será selecionado.

Figura 22 – Placa selecionada e botão de instalação dentro do *software*

Fonte: Autor

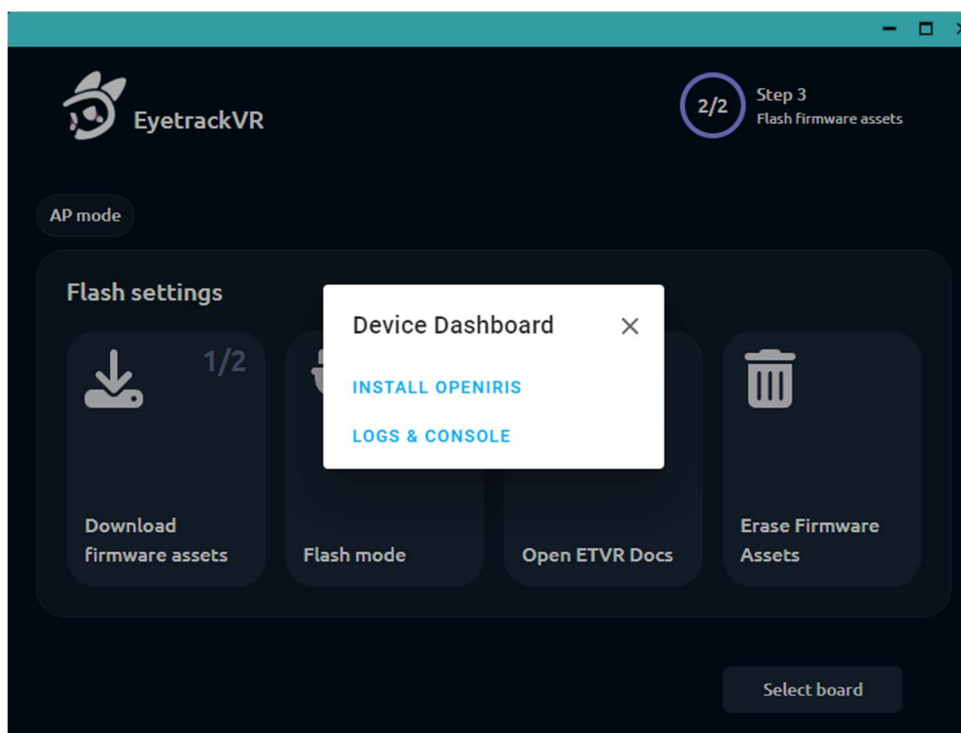
Por fim, novas opções serão apresentadas no *software* (Figura 23) e devem ser clicadas na seguinte ordem: *Download firmware assets*, para fazer o download dos arquivos necessários para a atualização e então *Flash mode*, para iniciar-se a confirmação da instalação.

Figura 23 – Novas opções presentes no *software*

Fonte: Autor

Após clicar-se na opção “*Flash mode*”, confirma-se a placa que deseja ser atualizada e então deve-se clicar em uma nova opção presente no *software* chamada “*INSTALL OPENIRIS*” (Figura 24), essa opção irá atualizar o *firmware* da placa microcontroladora.

Figura 24 – Opção *INSTALL OPENIRIS* presente no *software*



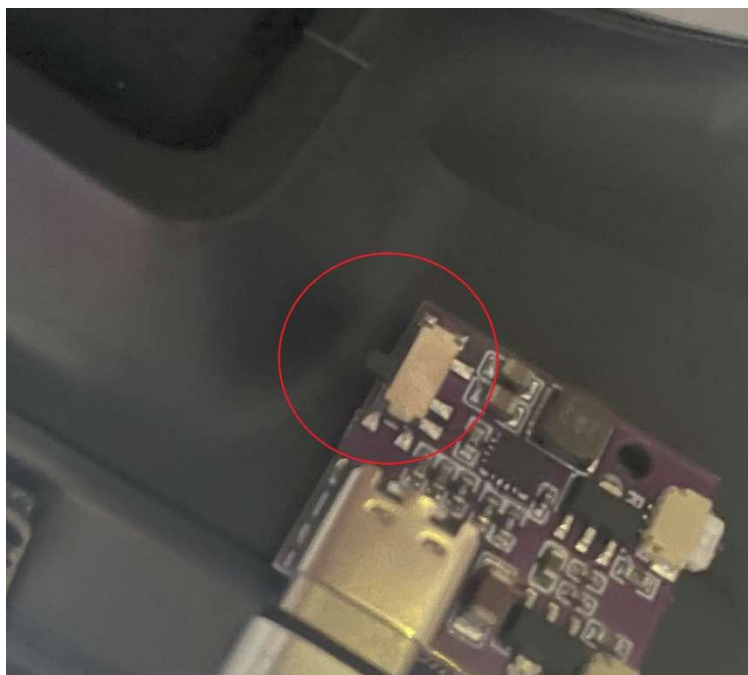
Fonte: Autor

Com isso, o processo de atualização do *firmware* das placas Seeed Studio XIAO ESP32-S3 estará concluído e toda a montagem e preparação do protótipo estará concluída.

3.3 Preparação do *software* para testes

Com a montagem e preparação do protótipo concluída, pode-se agora conectar o dispositivo de RV no computador e ligar um pequeno interruptor presente na placa *V4 mini Fully Solder-less LED Kit* (Figura 25, página seguinte), esse irá permitir a placa a alimentar os emissores de IR presentes dentro do dispositivo.

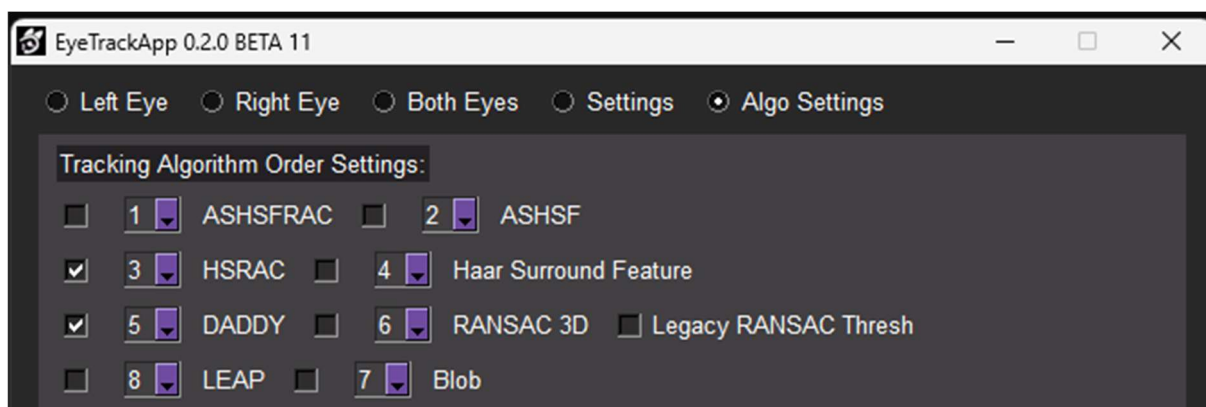
Figura 25 – Interruptor para ligar os emissores de IR



Fonte: Autor

Com o dispositivo de RV alimentado e os emissores de *IR* em funcionamento, deve-se executar o *software EyeTrackVR*, disponibilizado pela ETVR, esse irá localizar as portas USB que estão sendo utilizadas pelas câmeras e conectar elas diretamente ao *software*.

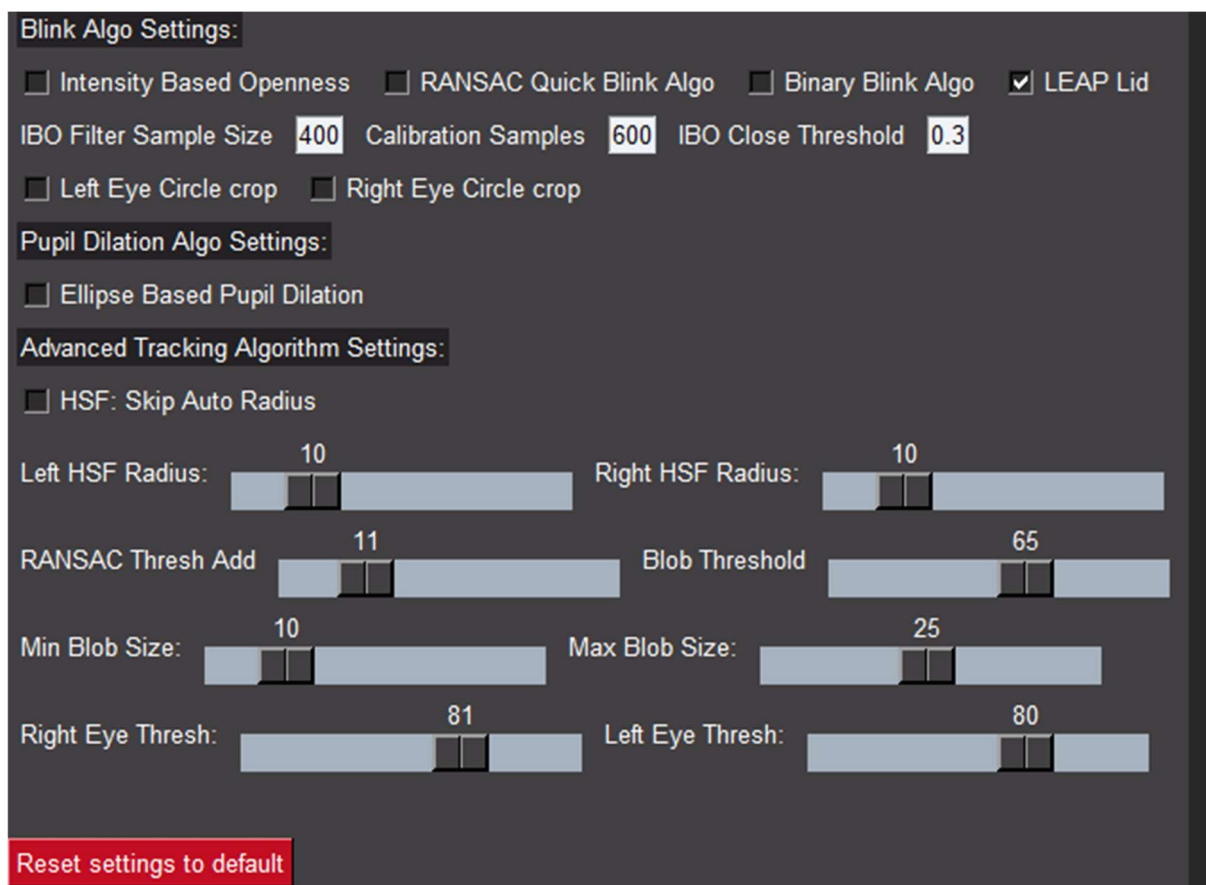
Após isso, deve-se navegar até a opção “*Algo Settings*” na parte superior da aba e selecionar os algoritmos que irá utilizar para o rastreamento das pupilas até encontrar-se o mais satisfatório para o usuário. No caso do protótipo montado, foram realizados alguns testes para verificar quais eram os mais precisos e o resultado foi deixar habilitados os algoritmos HSRAC e DADDY, como demonstrado na Figura 26.

Figura 26 – Algoritmos selecionados no *software*

Fonte: Autor

Além disso, foi-se ajustadas as configurações “*Intensity Basde Openness*” e “*Ellipse Based Pupil Dilation*” dentro do *software* (Figura 27), permitindo que mais movimentos sejam captados pelo mesmo e entreguem um *tracking* mais satisfatório.

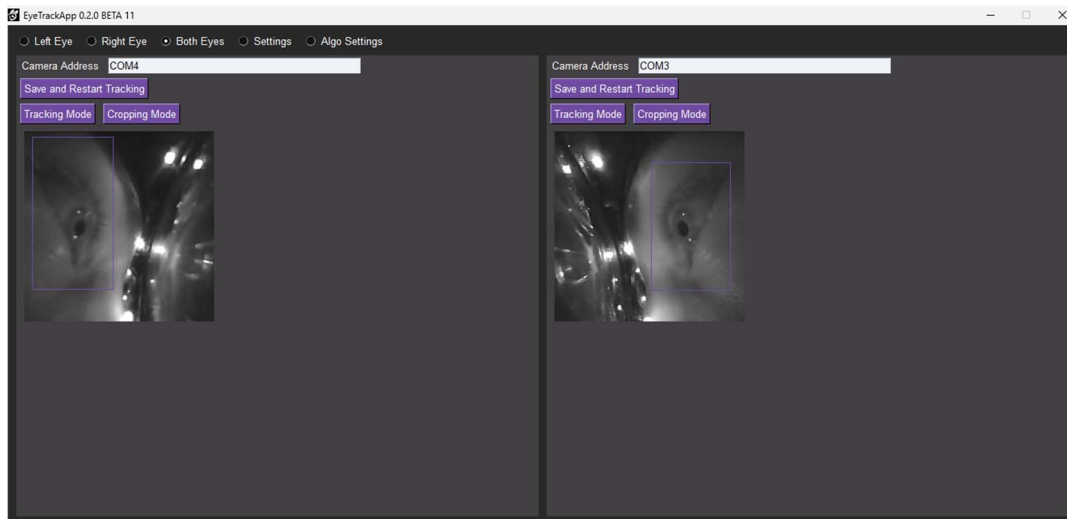
Figura 27 – Configurações ajustadas no *software*



Fonte: Autor

Por fim, volta-se para a opção “*Both Eyes*” selecionada a partir da parte superior da aba e tem-se a visão das câmeras presentes no protótipo. A partir dessa opção, seleciona-se a configuração “*Cropping Mode*” e então faz-se uma seleção quadrada ao redor do olho do usuário, tentando encaixar completamente o mesmo dentro da seleção (Figura 28, página seguinte).

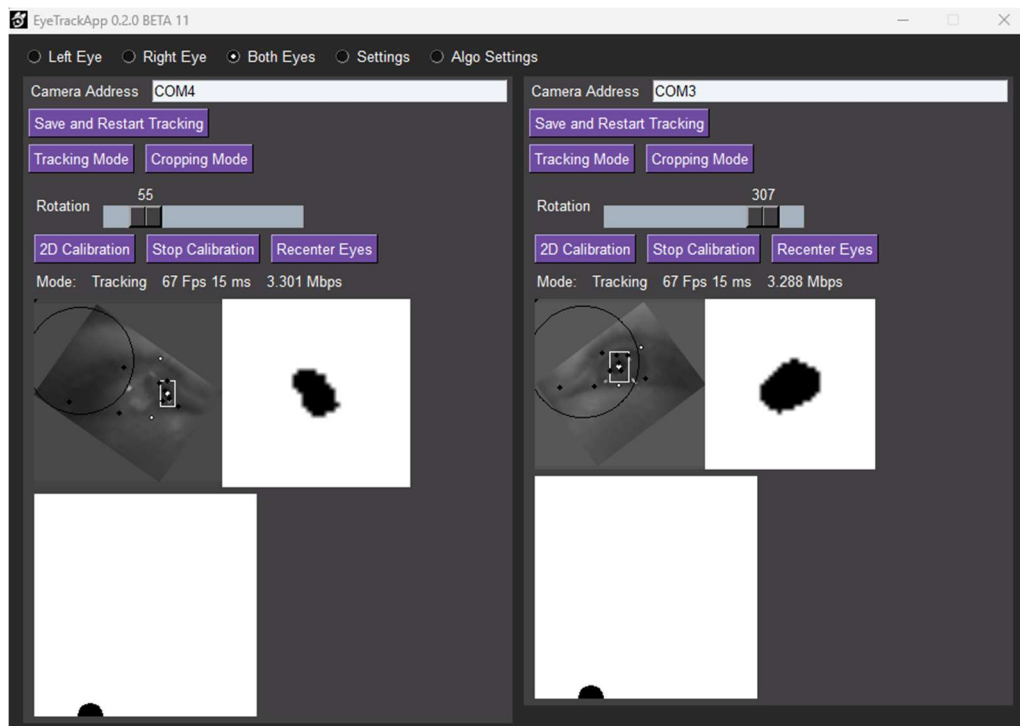
Figura 28 – Seleção dos olhos no modo “*Cropping Mode*”



Fonte: Autor

Após configurar a posição dos olhos, seleciona-se a opção “*Tracking Mode*” no *software* e seleciona-se a opção “*Recenter Eyes*” (Figura 29), isso irá mostrar um círculo roxo, dentro do dispositivo de RV, que deve ser encarado para calibrar a posição central dos olhos.

Figura 29 – Menu do aplicativo com a opção “*Recenter Eyes*”



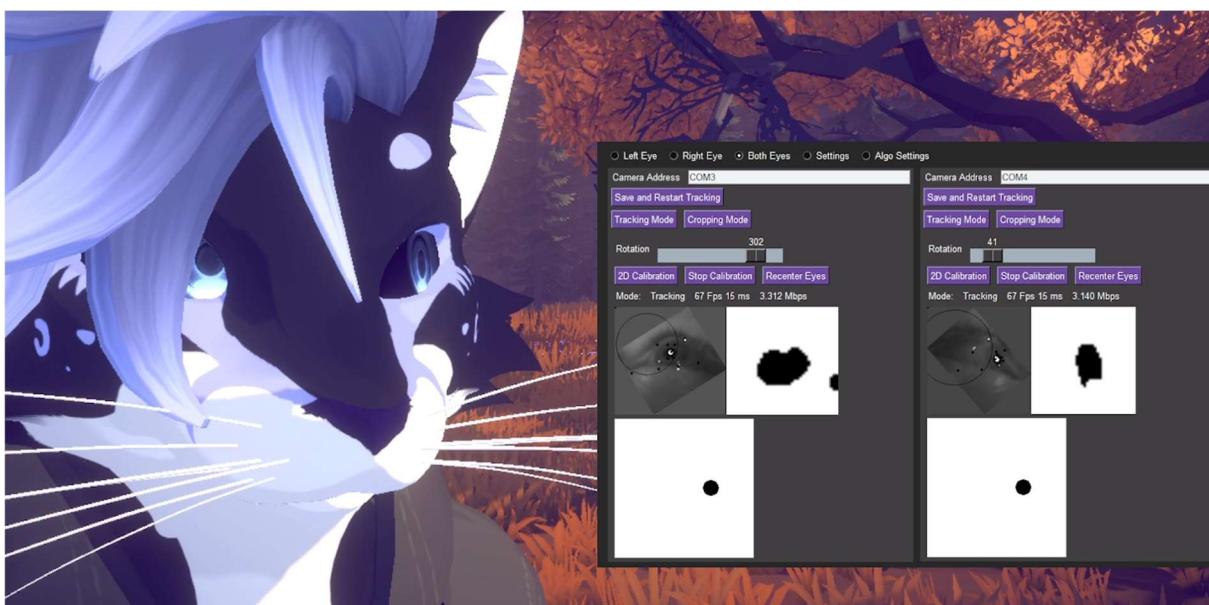
Fonte: Autor

Com isso, o *software* estará configurado para testes, podendo-se seguir para a etapa final do projeto.

3.4 Testes

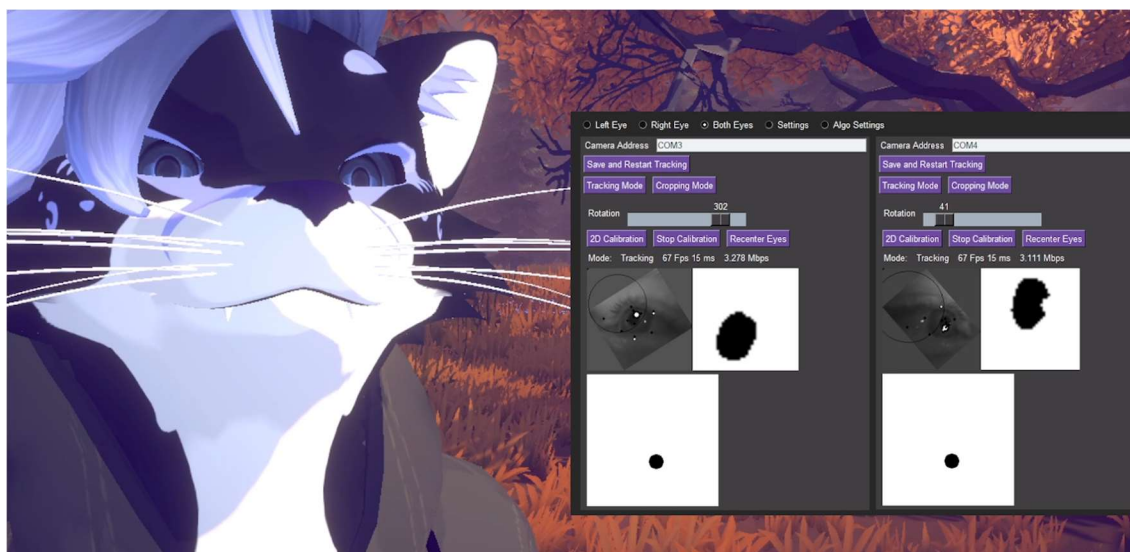
Como definido anteriormente, para o processo de testes irá ser executado o jogo *VRChat* e o aplicativo *EyeTrackVR* será capaz de se comunicar com o mesmo, alinhando o movimento do olho do personagem utilizado com o captado pelas câmeras presentes dentro do dispositivo de RV, como demonstrado nas figuras 30, 31 (página seguinte) e 32 (página seguinte).

Figura 30 – Personagem alinhando o movimento ocular para a esquerda com o usuário



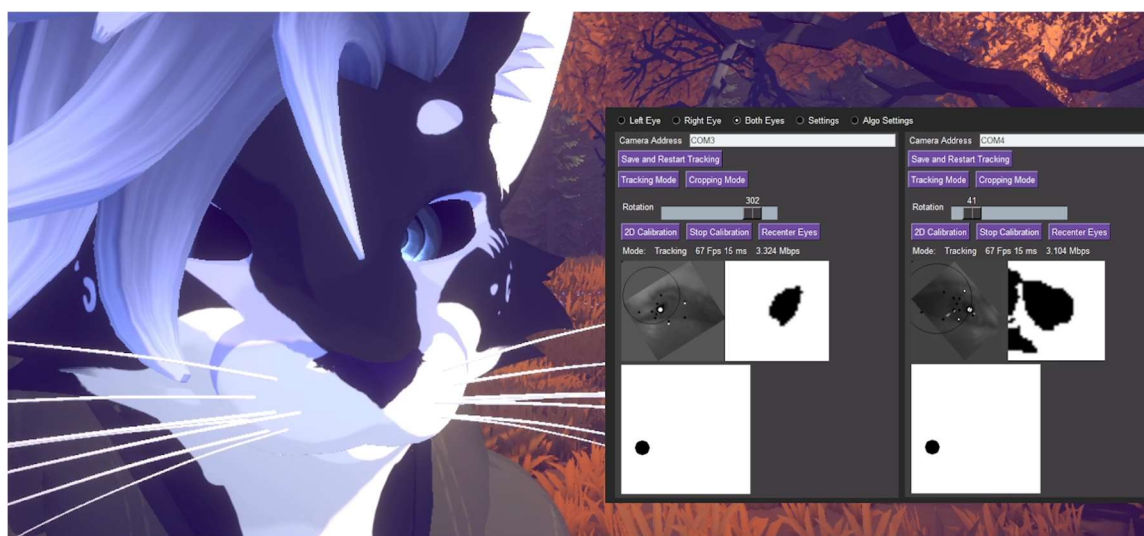
Fonte: Autor

Figura 31 – Personagem alinhando o movimento ocular para frente com o usuário



Fonte: Autor

Figura 32 – Personagem alinhando o movimento ocular para a direita com o usuário



Fonte: Autor

Como pode ser observado, o quadrado presente no aplicativo do rastreamento ocular trata-se de um sistema de coordenadas X e Y, onde o ponto preto representa a localização da pupila. Esse é traduzido diretamente ao jogo *VRChat* como uma rotação, rotacionando o olho do personagem na direção que o usuário está olhando.

4 Resultados

O protótipo desenvolvido demonstrou viabilidade técnica e funcionalidade na implementação de um sistema de *eye-tracking* para dispositivos de realidade virtual que não possuem suporte nativo. Através do uso de emissores de luz infravermelha e câmeras acopladas a um dispositivo de RV Valve Index, foi possível capturar movimentos oculares e integrar esses dados em tempo real com aplicações compatíveis, como o jogo *VRChat*.

Os testes realizados evidenciaram um alinhamento preciso entre os movimentos do usuário e a resposta visual do personagem virtual, validando a eficácia do protótipo. Além disso, a calibração das configurações no *software EyeTrackVR* permitiu personalizações para aprimorar a experiência do usuário, como ajustes de intensidade e algoritmos de rastreamento, garantindo uma usabilidade robusta.

O protótipo também destacou a possibilidade de adaptar tecnologias emergentes a dispositivos de gerações anteriores, ampliando o acesso a funcionalidades avançadas de forma econômica.

5 Conclusão

Este estudo demonstra que a implementação de um sistema de *eye-tracking DIY* é uma solução tecnicamente viável e economicamente acessível para dispositivos de realidade virtual que não possuem suporte nativo para essa tecnologia. Além de comprovar sua funcionalidade, o trabalho evidenciou como essa solução pode ser integrada a dispositivos existentes para substituir comandos manuais, simplificando e automatizando interações. Tal avanço reforça o potencial do *eye-tracking* como uma ferramenta poderosa para aprimorar a experiência de imersão e usabilidade em ambientes virtuais, democratizando o acesso a tecnologias de ponta.

A aplicabilidade do sistema transcende o escopo inicial de dispositivos de RV, abrindo novas possibilidades no campo de tecnologias assistivas. Ao permitir o controle de dispositivos apenas com os movimentos oculares, o sistema tem o potencial de revolucionar a acessibilidade para pessoas com necessidades especiais, oferecendo alternativas práticas e intuitivas para interações com softwares e dispositivos eletrônicos. Essa aplicação pode facilitar a realização de atividades cotidianas, como navegação em menus digitais, controle de cadeiras de rodas automatizadas, ou até mesmo o uso de computadores e smartphones sem a necessidade de controles manuais.

Outro aspecto promissor da pesquisa é a sua capacidade de expansão para outras partes do corpo. O desenvolvimento de sistemas de reconhecimento facial para a área da boca, por exemplo, poderia complementar o rastreamento ocular, oferecendo novas formas de interação baseadas em expressões faciais ou movimentos labiais. Essa abordagem não apenas substituiria comandos manuais, mas também possibilitaria uma comunicação mais natural e intuitiva com os sistemas digitais. O uso combinado dessas tecnologias ampliaria significativamente o alcance das aplicações, incluindo cenários como a comunicação assistida por movimentos faciais para pessoas com limitações motoras severas.

Além disso, a modularidade do sistema apresentado neste estudo facilita sua adaptação a diferentes modelos de dispositivos, tanto atuais quanto de gerações anteriores, aumentando sua relevância e aplicabilidade. O desenvolvimento de *hardware* e *software* personalizáveis também permite que futuros projetos explorem algoritmos mais sofisticados para aprimorar a precisão do rastreamento e expandir as funcionalidades, como análise comportamental e interação em realidade aumentada.

Futuras pesquisas podem se beneficiar dos resultados apresentados aqui ao explorar novas áreas de aplicação, como o uso em ambientes educacionais, simuladores de treinamento, ou mesmo em dispositivos de uso médico e terapêutico. A ampliação do conceito para outros contextos e a otimização de seu desempenho têm o potencial de consolidar essa tecnologia como uma solução universal e inclusiva, capaz de transformar tanto a maneira como interagimos com dispositivos digitais quanto o acesso a tecnologias assistivas.

Em suma, o presente estudo não apenas valida a viabilidade técnica de um sistema *DIY* de *eye-tracking* para RV, mas também destaca suas possibilidades transformadoras em diferentes cenários, seja para aprimorar tecnologias existentes ou para criar novas soluções voltadas à inclusão e acessibilidade.

Referências

- ACGIH. *Threshold Limit Values for Chemical Substances and Physical Agents & Biological Exposure Indices*. Cincinnati: American Conference of Governmental Industrial Hygienists, 2001.
- AXELSON, Jan. *USB complete: the developer's guide*. Lakeview research LLC, 2015.
- BANZI, Massimo; SHILOH, Michael. *Getting started with Arduino*. Maker Media, Inc., 2022.
- BARRETO, Ana Margarida. *Eye tracking como método de investigação aplicado às ciências da comunicação*. Revista Comunicando, v. 1, n. 1, p. 168-186, 2012.
- BURDEA, Grigore C.; COIFFET, Philippe. *Virtual reality technology*. John Wiley & Sons, 2003.
- CHUA, Chee Kai; LEONG, Kah Fai. *3d Printing and additive Manufacturing: Principles and applications-of rapid Prototyping*. World Scientific Publishing Company, 2016.
- CLAY, Viviane; KÖNIG, Peter; KOENIG, Sabine. *Eye tracking in virtual reality*. Journal of eye movement research, v. 12, n. 1, 2019.
- CRAFT, Brock. *Arduino projects for dummies*. John Wiley & Sons, 2013.
- FREINA, Laura; OTT, Michela. *A literature review on immersive virtual reality in education: state of the art and perspectives*. In: The international scientific conference elearning and software for education. 2015. p. 10-1007.
- GEVORGYAN, Menua; MAMIKONYAN, Arsen; BEYELER, Michael. *OpenCV 4 with Python Blueprints* – Second Edition. Packt, 2020.
- GIBSON, Ian. *Additive manufacturing technologies 3D printing, rapid prototyping, and direct digital manufacturing*. 2015.
- INTERNATIONAL COMMISSION ON NON-IONIZING RADIATION PROTECTION et al. *ICNIRP guidelines on limits of exposure to incoherent visible and infrared radiation*. Health Physics, v. 105, n. 1, p. 74-96, 2013.
- PHYSICS-DUDE. **Phys-Index-EyetrackVR-HW**. Disponível em: <https://github.com/Physics-Dude/Phys-Index-EyetrackVR-HW/tree/main>. Acesso em: 20 jun. 2024.

PROHURTZ; SUMMER; LOROW; DAOFFICIALWIZARD. *EyeTrackVR Docs: Open-source and affordable VR eye tracking*. Disponível em: <https://docs.eyetrackvr.dev>. Acesso em: 20 jun. 2024.

RODRIGUES, Gessica Palhares et al. **Realidade virtual: conceitos, evolução, dispositivos e aplicações**. Interfaces Científicas-Educação, v. 1, n. 3, p. 97-109, 2013.

SHERMAN, William R.; CRAIG, Alan B. *Understanding virtual reality*. San Francisco, CA: Morgan Kauffman, 2003.