



**POSITIONING AND DISPLAY OF 3D IMAGES USING AUGMENTED
REALITY GLASSES FOR SURGICAL APPLICATION**

Research Internships Abroad Proposal destined to UNIVERSITY NAME HERE (SIGLA).

FAPESP Process Number: 2020/15835-4

Scholarship Supervisor: Dr. Glauco Augusto de Paula Caurin

Supervisor for BEPE's Internship: PROFESSOR NAME HERE

Student: Calvin Suzuki de Camargo

General Internship Proposal information

- Project title:

Positioning and display of 3D images using augmented reality glasses for surgical application

- Scholarship Supervisor:

Dr. Glauco Augusto de Paula Caurin

- Supervisor for BEPE's Internship:

PROFESSOR NAME HERE

- Student:

Calvin Suzuki de Camargo

- FAPESP Process Number:

2020/15835-4

- Home institution:

São Carlos School of Engineering da University of São Paulo

- Proposed Duration:

March 2021 to 28 February 2022

Abstract

This is an internship proposal that aims to study the application of smart glasses with augmented reality, which can serve as an auxiliary device in surgical applications. For this, the project deepened concepts of computer vision and computer graphics with the integration between Unity and OpenCV. A system architecture was defined that allows the superimposition of a 3D model on a patient's head, indicating electrode implantation points during the neuro-surgical procedure guided by stereoelectroencephalography (SEEG). This architecture was based on communication between the computer and the glasses over the local network: The computer receives the images from the camera on the glasses and uses the detection method using ArUco fiducial markers to estimate the patient's position and sends the coordinates to the glasses, indicating where the projection must be displayed. The partial results of the application show a fast responsiveness with the chosen estimation method.

Palavras-chaves: Smart glasses, Augmented reality, Surgical applications, Computer vision.

Table of Contents

General project information	i
Abstract	ii
1 Introduction	1
1.1 Motivation	1
1.2 Objective	2
1.3 Brief history of the project	3
2 Background	4
2.1 Ângulos de Euler	4
2.2 Rotação de Rodrigues	5
2.3 Calibração	5
2.3.1 Rotação	6
2.3.2 Translação	6
2.3.3 <i>Field of View</i>	7
2.3.4 Distorção	8
2.4 Desenvolvimento Unity	9
2.4.1 Ajuste dos intrínsecos	10
3 Proposta de extensão	11

3.1	Objetivo	12
3.2	Metodologia	12
3.3	Cronograma	14
4	Participação de eventos acadêmicos	15
	References	16

1 Introduction

1.1 Motivation

Currently in medicine, surgeries have the objective of causing the least aggression to the patient's organism, which implies a shorter hospitalization time; less incidence of postoperative complications; less pain; and quicker recovery. However, among the many challenges involved in its implementation, the physician's visual ability, which involves the ability to locate and identify sensitive tissues, is essential to reduce damage to the patient during a surgical procedure, leading us to the question: How can we enhance the doctor's vision to increase the quality of these operations?

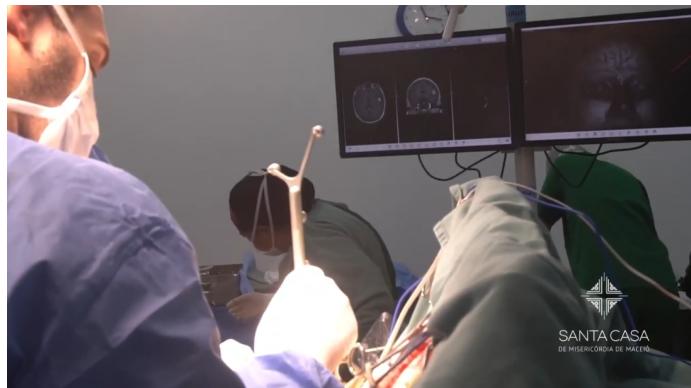


Figura 1.1: Example of using a neuronavigator in surgery. Source: (SERVIÇO..., 2019).

To this end, software that informs and assists the doctor in real time during the operation is a great way to increase the chances of success in these surgeries. Currently, these software are very present in the planning and execution of operations, however, neuronavigation systems are an example, whose function is to allow the visualization, in a precise way, of the patient's brain structures. The neurosurgeon has to turn his head to read the monitors and loses focus on the activity being conducted, collaborating for a longer operation time and favoring the exhaustion of the professional, in a way that increases his potential for errors during surgical procedures (Figure 1.1).

To do this, augmented reality (AR) technology offers the potential to reduce these li-

mitations, as three-dimensional figures can superimpose the doctor's vision to facilitate visualization and location of the objectives of the operation while keeping their eyes focused on the patient (CHO et al., 2020). What was done in this project is a research involving computer vision and computer graphics in the field of AR and an application for the glasses of the company Seiko Epson Corporation®, model Moverio BT-350™. Receiving physical and technical support from the Aeronautical Technology Laboratory (Aerotech) team at the University of São Paulo, this study corroborates the group's objective, which is to develop a collaborative neurosurgical multi-application platform.

1.2 Objective

Display information and position three-dimensional models in a region of space with AR, in a way that facilitates the surgeon's access to information during neurosurgery; study and record the response of the equipment used in terms of graphic quality and system response latency. All of this, with the central objective of increasing the proximity of the surgeon to the AR technology as a support during surgical procedures.

1.3 Brief history of the project

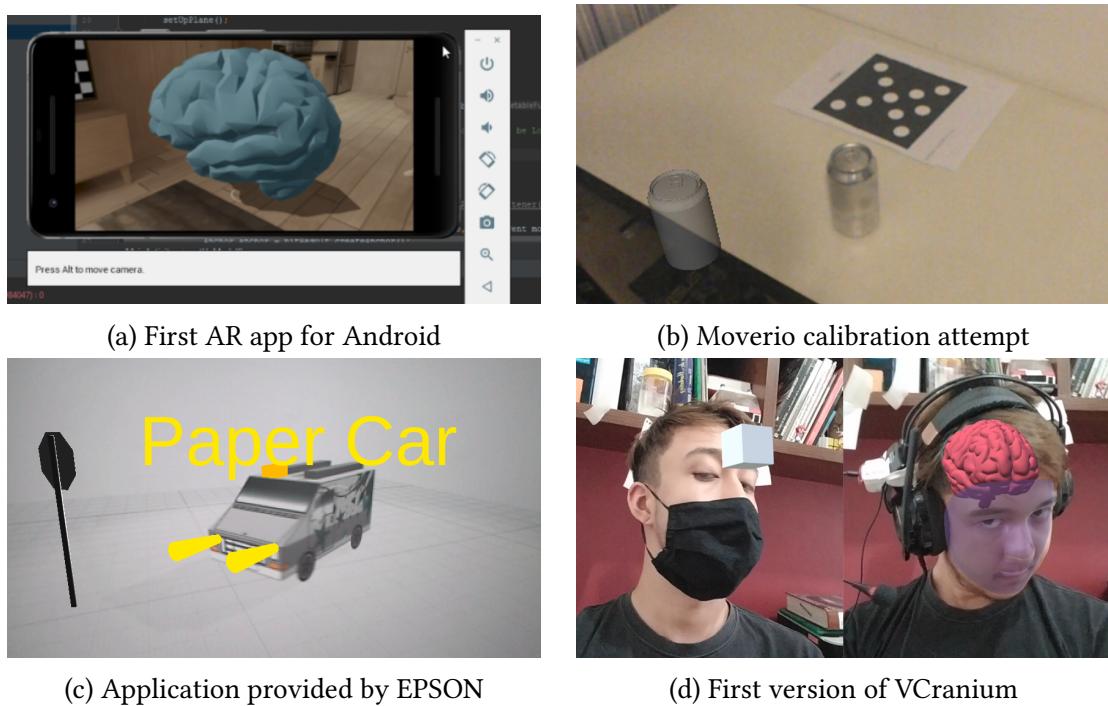


Figura 1.2: History of research since 2020.

Since the beginning of work on the project in 2020, different types of contact have been experienced with the development of software for the Android operating system (Figure 1.2a); tests of the Moverio BT-350 augmented reality glasses documentation tools (Figures 1.2b and 1.2c); and the creation of applications that illustrate the purpose of VCranium¹ (Figure 1.2d).

¹Fantasy name assigned to the project of positioning system and display of augmented reality for surgical application

2 Background

Durante os estudos de transformações de rotação e translação de objetos em espaço \mathbb{R}_3 , o desenvolvimento do projeto fundamentou-se em outros métodos além das disciplinas de Geometria Analítica e Álgebra Linear. Nesse estudo, foram definidas a rotação por ângulos de Euler e a rotação de Rodrigues:

2.1 Ângulos de Euler

É um método que descreve a orientação de corpos rígidos por meio de três rotações sucessivas, desenvolvido no século XVIII por Leonard Euler. Consiste em realizar três multiplicações matriciais, introduzidas a seguir (EULER..., 2023).

$$A = B.C.D = \text{Rot}Z(\phi).\text{Rot}X(\theta).\text{Rot}Z(\psi)$$

$$A = \begin{bmatrix} \cos \phi & \sin \phi & 0 \\ -\sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

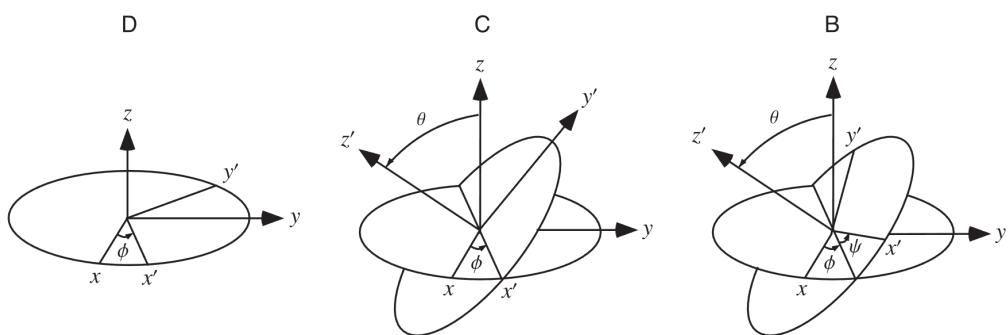


Figura 2.1: Sequência de rotações ZXZ de Euler. Fonte: (EULER..., 2023).

Com isso obteremos uma matriz ortogonal A que será a transformação do objeto em \mathbb{R}_3 ou no mundo virtual do *Unity*. Esse método de rotação é utilizado no *Unity* como uma das propriedades de um objeto em relação a outro objeto (*parent*). Neste, as rotações podem ser

configuradas por ângulos de Euler, mas internamente *Unity* traduz o método para quatérnios¹ com o fim de evitar *Gimbal Locks*² e otimizar os cálculos em cada iteração (UNITY..., 2023).

2.2 Rotação de Rodrigues

É considerado um método muito eficiente de descrever uma rotação, desenvolvido em 1989 por Olinde Rodrigues sobre os fundamentos de Leonhard Euler. O método aplica uma rotação no vetor \vec{v} no eixo de rotação descrito com o vetor \vec{k} e um ângulo θ (RODRIGUES..., 2023):

$$\vec{v}_{rot} = \vec{v} \cos \theta + (\vec{k} \times \vec{v}) \sin \theta + \vec{k} (\vec{k} \cdot \vec{v}) (1 - \cos \theta)$$

A biblioteca do *OpenCV* utiliza a descrição compacta de Rodrigues para detectar os marcadores *ArUco*. No entanto, invés de retornar o vetor \vec{k} e o ângulo θ , ele retorna $\vec{r} = \vec{k} \cdot \theta$. Como \vec{k} é um vetor unitário, podemos obter o ângulo com o cálculo do módulo de \vec{r} : $\theta = |\vec{r}|$.

2.3 Calibração

O processo de calibração do *VCranium* foi complexo por abranger diversos conceitos de visão computacional e sua aplicabilidade na plataforma do *Unity* (UNITY..., 2021). Decidimos fragmentar o problema e estudá-los separadamente:

Rotação: Obtido com o *RVEC* que porta o vetor \vec{r} explicado na Seção 2.2 para cada *frame*;

Translação: Obtido com o *TVEC* em forma de um vetor de translação de cada *frame*;

Field of View: (*FOV*) Calculado com as informações intrínsecas da calibração da câmera;

Distorção: Calculado com as informações de distorção da calibração da câmera;

¹Os quatérnios foram desenvolvidos pelo matemático irlandês William Rowan Hamilton em 1843 e aplicados à mecânica no espaço tridimensional. São amplamente utilizados em computação gráfica por simplificar a quantidade de operações matemáticas de rotações em relação a outros métodos (QUATERNION..., 2023).

²É uma condição em que perde-se um grau de liberdade, degenerando a rotação do sistema para bidimensional. Em engenharia, esse problema se caracteriza como uma singularidade e afeta a previsibilidade de diversos sistemas (GIMBAL..., 2023).

2.3.1 Rotação

A informação da rotação do modelo virtual do cérebro é adquirida da comunicação *socket* com o servidor em *Python*. Abaixo está estruturada o formato da mensagem enviada a cada iteração do computador para a aplicação:

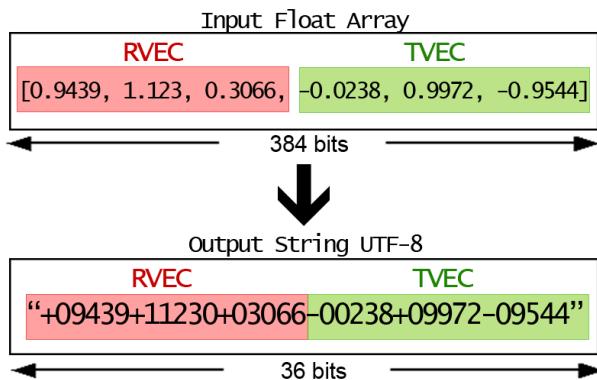


Figura 2.2: Estrutura da mensagem via *socket*. Fonte: Autor.

Um método foi implementado para formatar cada mensagem gerada pela detecção da posição do marcador, transformando uma lista do *Python* em uma *string* com o formato de *UTF-8*, como é mostrado no *RVEC* da Figura 2.2. Assim que a aplicação recebe os três números, eles representam o vetor de rotação com as coordenadas X, Y e Z, respectivamente. Para descobrirmos a magnitude da rotação neste eixo, deve-se calcular o módulo desse vetor de rotação, o resultado é dado em radianos.

2.3.2 Translação

Da mesma forma que a rotação, a translação é recebida do *TVEC* da mensagem da Figura 2.2. Nesta, a informação é recebida em milímetros, portanto, devemos ajustar a escala da translação manualmente para corrigir a projeção do marcador no mundo virtual do *Unity*. Com a rotação e translação completas, temos a calibração extrínseca da projeção. Assim como foi explicado no último relatório, a projeção deve estar visualmente boa, especialmente nas regiões centrais da tela, representado na Figura 2.3.

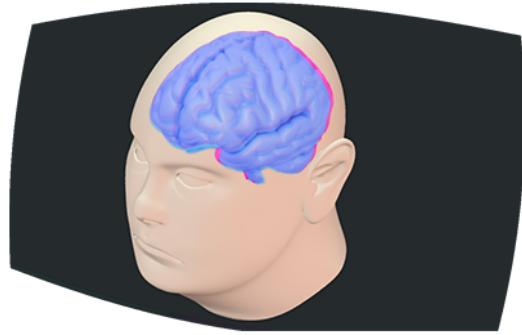


Figura 2.3: Exemplo de sobreposição sem calibração intrínseca. Fonte: Autor.

2.3.3 *Field of View*

O *Field of View* (*FOV*) é calculado da calibração da câmera com o *chessboard* que calcula a distorção e os intrínsecos da câmera, nomeados respectivamente de *DIST* e *MTX*. Para determinar o *FOV* da câmera observamos a matriz *MTX* (OPENCV..., 2023):



Figura 2.4: Calibração dos intrínsecos da câmera com o *chessboard*. Fonte: Autor.

$$MTX = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Utilizamos as distâncias focais da câmera, f_x ou f_y , e a informação das dimensões da imagem em *pixels* para obter o *FOV* em X e Y:

$$FOV_x = 2 \arctan \left(\frac{w}{2f_x} \right); \quad FOV_y = 2 \arctan \left(\frac{h}{2f_y} \right)$$

Sendo w e h a largura e a altura em *pixels*, respectivamente. Não existe diferença em utilizar qualquer um dos dois, pois o *Unity* suporta ambas as configurações no objeto da câmera virtual.

2.3.4 Distorção

A documentação do *OpenCV* baseia-se na câmera *pinhole* e modela duas principais distorções: radial e tangencial (PINHOLE..., 2023).

Radial: A distorção radial é percebida como retas se curvando em relação ao centro. Causada pela fabricação da forma circular das lentes, o que contribui para um maior campo de visão.

$$x_{distorted} = x(1 + k_1r^2 + k_2r^4 + k_3r^6)$$

$$y_{distorted} = y(1 + k_1r^2 + k_2r^4 + k_3r^6)$$

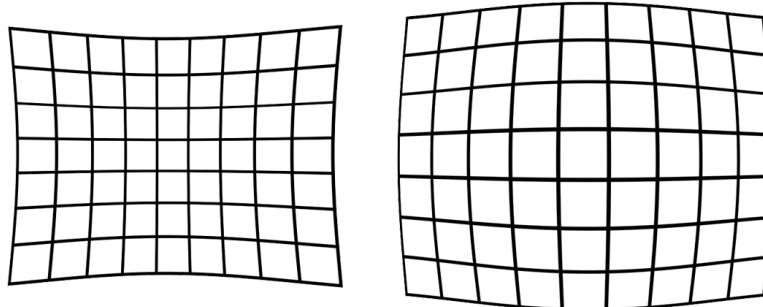


Figura 2.5: Exemplos de distorção radial. Fonte: Autor.

Tangencial: A distorção tangencial é percebida em regiões da imagem que parecem mais próximas ou mais distantes da câmera de forma não uniforme. Causada pelas tolerâncias de paralelismo da montagem do sensor com as lentes internas do equipamento.

$$x_{distorted} = x + 2p_1xy + p_2(r^2 + 2x^2)$$

$$y_{distorted} = y + p_1(r^2 + 2y^2) + 2p_2xy$$

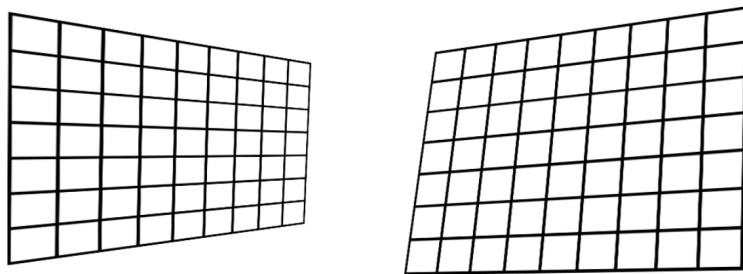


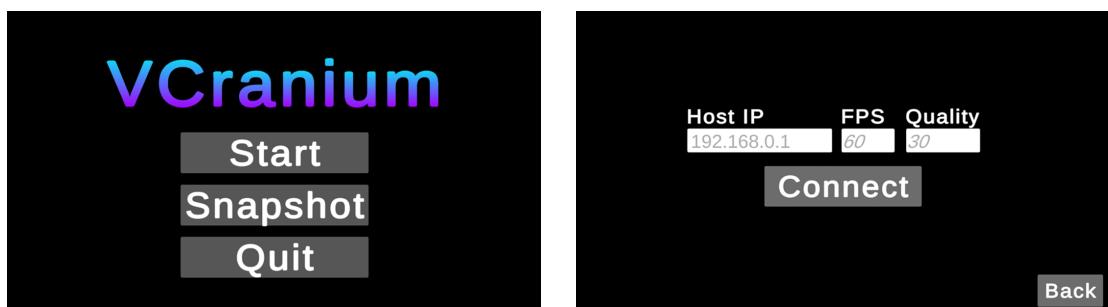
Figura 2.6: Exemplos de distorção tangencial. Fonte: Autor.

Utilizando as fórmulas apresentadas acima e os parâmetros intrínsecos f_x f_y c_x c_y e distorção k_1 k_2 k_3 p_1 p_2 , tem-se todas as variáveis necessárias para a modelagem da câmera com *OpenCV* (OPENCV..., 2023):

$$DIST = [k_1 \ k_2 \ p_1 \ p_2 \ k_3]; \quad MTX = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix};$$

2.4 Desenvolvimento Unity

Dada a teoria da visão computacional apresentada na Seção 2.3, traz-se para a aplicação no ambiente de desenvolvimento *Unity*. Para isso, deve-se organizar de "cenas", onde o usuário pode transitar utilizando a interface, como na Figura 2.7, e utilizar a linguagem C# como *back-end* para a criação de *scripts*.



(a) Menu principal do programa

(b) Tela de conexão com computador

Figura 2.7: Imagens da interface do *VCranius*. Fonte: Autor.

2.4.1 Ajuste dos intrínsecos

A Seção 2.3 também explicita que obtemos todos os parâmetros intrínsecos e de distorção de uma câmera após a calibração com o método do *chessboard* (Figura 2.4). Porém, é necessário um método que realize a calibração da câmera em que o *VCranium* está sendo executado, para isso, deve-se criar um método auxiliar que capture uma série de imagens e envie para o computador calibrar e encontrar os parâmetros (OPENCV..., 2023).

A principal ideia que se tem ao criar esse sistema é receber uma imagem assim que o usuário pressionar o botão de captura. Assim como foi estudado em relatórios anteriores, é necessário construir um *header* simples que antecede o envio da imagem para o servidor preparar a sua recepção, e então, guardar a informação de cada imagem em uma lista auxiliar. O próximo passo é usar o comando *calibrateCamera* da biblioteca do *OpenCV* e o programa retorna as informações necessárias (f_x , f_y , c_x , c_y , k_1 , k_2 , k_3 , p_1 , p_2) (OPENCV..., 2023).

O desenvolvimento desse programa auxiliar não apresenta maiores desafios se comparado com o histórico do projeto desde 2021. No entanto, assim que o aplicativo *Unity* receber todos os parâmetros, não é trivial a distorção do modelo para a sobreposição em uma aplicação de realidade do tipo *VST* (*Video See-Through*).

Visto que seria necessário um estudo mais aprofundado no assunto de calibração, elaboração de conceitos da disciplina de Computação Gráfica para a vista em perspectiva, e a aplicação desse trabalho seriam restritos somente ao modelo *VST*, foi decidido privar a calibração intrínseca do projeto no dado momento. A Figura 2.3 mostra que ao fim da calibração extrínseca, a sobreposição é visualmente boa, i.e., em boa parte de ilustração, o erro da sobreposição é insignificante e, com isso, não apresenta problemas para demonstração de conceitos.

3 Proposta de extensão

No decorrer dos dois anos de desenvolvimento do projeto, buscou-se a formação no campo de desenvolvimento de aplicações, comunicação via *socket*, aprendizado de *Unity*, aprofundamento em visão computacional e, principalmente, pesquisa bibliográfica em aplicação de realidade aumentada em neurocirurgias. Contudo, toma-se a abordagem da aplicação dos óculos *Moverio BT-350* como ferramenta auxiliar de visualização anatômica do paciente em tempo real durante uma neurocirurgia. Para isso, foi proposto uma arquitetura compatível com equipamentos disponíveis e com tempo de resposta adequado.

No momento, o projeto funciona da maneira esperada, porém com muito espaço para ser aperfeiçoado antes de ser levado para a avaliação dos médicos. Os quesitos que podem ser melhorados são:

Método da Detecção de Posição: O projeto utiliza somente um marcador *ArUco* para estimar sua posição, o que traz principalmente o problema do usuário perder o marcador de vista por causa do ângulo de observação.

Otimização da latência do sistema: Os fatores que limitam o tempo de resposta do sistema é principalmente a conexão sem fio dos óculos com o *access point* e do roteador até o servidor.

A solução desses fatores torna a pesquisa apresentável para os médicos do Centro de Cirurgia de Epilepsia (CIREP) do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto-USP. A parceria aproxima dos integrantes do laboratório dos pesquisadores do CIREP em forma de reuniões *online* e eventuais visitas até o hospital das clínicas. Seguindo essa oportunidade, se forem preparadas demonstrações do *VCranium*, o projeto pode receber sugestões diretas dos médicos e compiladas em um artigo científico.

3.1 Objetivo

Otimizar os processos do sistema de reconhecimento de posição da cabeça do paciente; definir um método de reconhecimento mais robusto e que permita demonstrações. Por fim, apresentar o projeto aos pesquisadores do Centro de Cirurgia de Epilepsia do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto-USP.

3.2 Metodologia

O projeto já possui uma arquitetura que permite a alteração da forma de final de reconhecimento da pose de uma cabeça humana. Dessa forma, alterar o método de detecção de *ArUco* pode ser feito sem problemas. A confiabilidade comprovada dos métodos dos marcadores pode ser ampliada com o uso de mais marcadores, criando uma maior cobertura de atuação do sistema.

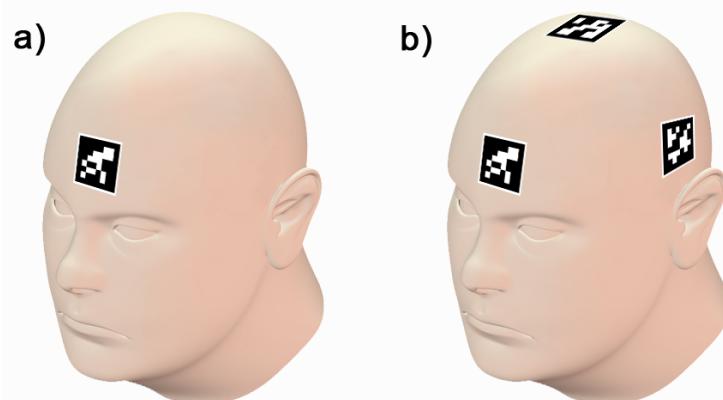


Figura 3.1: a) Aplicação atual de marcadores no projeto. b) Exemplo de utilização de múltiplos marcadores. Fonte: Autor.

Atualmente, o projeto aplica os marcadores como na Figura 3.1a. Dessa forma, o sistema está limitado a estimar a posição da cabeça somente em sua face frontal, i.e., não identifica as laterais e face traseira da cabeça. Uma solução para esse problema é proposto na Figura 3.1b, que invés de um único marcador, múltiplos marcadores são colocados em todas os lados da cabeça, com a de reconhecer a sua posição independentemente em cada marcador.

Uma forma de posicionar múltiplos marcadores é utilizar uma "tiara" para portá-los, um esboço desse objeto foi construído com papel na Figura 3.2. Esse método utiliza transformações de base, fundamentados da disciplina de Geometria Analítica e Álgebra Linear, para a estimar a

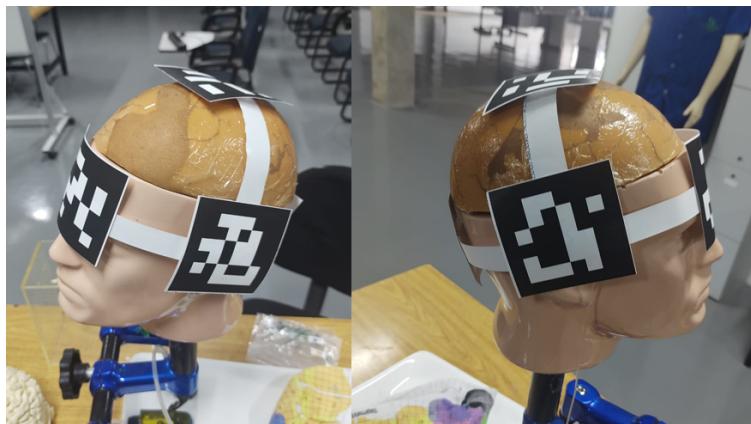


Figura 3.2: Esboço de uma tiara que suporta um marcador em cada lado da cabeça. Fonte: Autor.

posição central da cabeça e sua orientação. Para isso, deve-se saber a posição e orientação de cada marcador em relação ao centro da cabeça e, visto que os tamanhos da cabeça variam, encontra-se a necessidade de uma calibração para cada paciente. Pensando na impossibilidade do contato direto do marcador com a pele do paciente e a possível oclusão do marcador por ferramentas e as mãos do médico, esse método se torna uma opção menos viável para situações cirúrgicas.

Uma solução alternativa seria utilizar múltiplos marcadores distantes do paciente, mas ainda visíveis na câmera dos óculos como na Figura 3.3. Os marcadores em formato de cubo facilitam a montagem a calibragem de cada marcador em relação ao centro da cabeça.

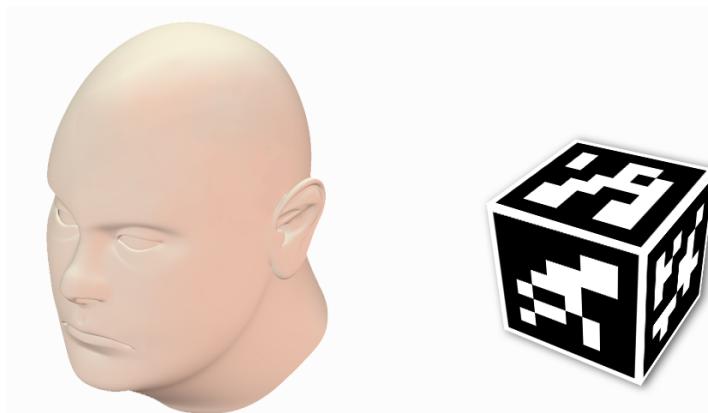


Figura 3.3: Solução de estimativa de posição com cubo de marcadores. Fonte: Autor.

Por fim, o projeto vai aproveitar o tempo de extensão para fazer um pedido de Bolsa Estágio de Pesquisa no Exterior (BEPE) para buscar cooperação com grupos de pesquisa do exterior. O intercâmbio, além de colocar a comunicação acadêmica em prática, tem a oportunidade de receber críticas aprofundadas e criativas no assunto de realidade aumentada, visto que muito desses equipamentos são raramente vendidos no Brasil.

3.3 Cronograma

O controle de atividades vai ser feito com encontros mensais e semanais com o coorientador, orientador e integrantes do laboratório. Um cronograma foi elaborado na tabela 3.1 para servir de guia para o progresso do projeto.

Atividade	Bimestre	Execução				
		1	2	3	4	5
Programação dos novos métodos com marcadores						
Montagem dos protótipos de tiara e cubo						
Estudos no exterior (BEPE)						
Produção de relatório final e relatório BEPE						
Apresentação final do trabalho*						

Tabela 3.1: *A apresentação para os pesquisadores da parceria do Centro de Cirurgia de Epilepsia (CIREP) do Hospital das Clínicas da Faculdade de Medicina de Ribeirão Preto-USP

4 Participação de eventos acadêmicos

Novamente, os docentes e discentes do laboratório Aerotech se orgulham de todos terem a oportunidade promover nossas pesquisas no 30º Simpósio Internacional de Iniciação Científica e Tecnológica da USP (SIICUSP). Dentre os integrantes, somente um deles foi convidado para a segunda fase que, felizmente, também foi condecorado com uma menção honrosa.

Durante a apresentação, o projeto recebeu algumas provocações que ampliaram as possibilidades de aperfeiçoamento em aspectos de otimização e aplicabilidade. Durante uma apresentação *online* foram respondidas as perguntas:

É possível mudar de Python para C/C++ para otimizar processo? Resposta: É possível, porém isso traz problemas no quesito da velocidade do desenvolvimento, disponibilidade de documentação, bibliotecas e ferramentas. No entanto, é uma ótima opção quando o projeto ter uma versão funcional para testes, tendo um potencial reduzir a latência de cada iteração.

O quanto longe o projeto está de um teste prático? Resposta: No momento, o projeto consegue fazer demonstrações básicas com as funções de detectar e sobrepor um modelo virtual de um cérebro um marcador em tempo real (aproximadamente 15 vezes por segundo). Porém, o projeto abre muitas possibilidades de ser aperfeiçoados e, por isso, é desejável um trabalho sobre o programa atual antes da apresentação final aos médicos.

As provocações dos avaliadores trouxeram a mensagem que o projeto ainda possui espaço para desenvolver seu algoritmo na questão da otimização e robustez que foram refletidos na Seção 3. Espera-se que a extensão do projeto tenha um perfil voltado a construção de ferramentas para testes práticos do sistema.

References

- CHO, J. et al. Enhancing Reality: A Systematic Review of Augmented Reality in Neuronavigation and Education. **World Neurosurgery**, p. 186–195, mai. 2020. DOI: 10.1016/j.wneu.2020.04.043.
- EULER Angles: Euler's rotation theorem. [S.l.: s.n.], 2023. <https://mathworld.wolfram.com/EulerAngles.html>. Acessado em: 25 de janeiro de 2023.
- GIMBAL Lock: Article about Gimbal Lock. [S.l.: s.n.], 2023. https://en.wikipedia.org/wiki/Gimbal_lock. Acessado em: 25 de janeiro de 2023.
- OPENCV Camera Calibration. [S.l.: s.n.], 2023. https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html. Acessado em: 25 de janeiro de 2023.
- PINHOLE Camera. [S.l.: s.n.], 2023. https://en.wikipedia.org/wiki/Pinhole_camera. Acessado em: 25 de janeiro de 2023.
- QUATERNION: Article about quaternions in mathematics. [S.l.: s.n.], 2023. <https://en.wikipedia.org/wiki/Quaternion>. Acessado em: 25 de janeiro de 2023.
- RODRIGUES rotation formula: Article about Rodrigues method in mathematics. [S.l.: s.n.], 2023. https://en.wikipedia.org/wiki/Rodrigues'_rotation_formula. Acessado em: 25 de janeiro de 2023.
- SERVIÇO de Neurocirurgia retira lesão cerebral com paciente acordado: Youtube. [S.l.: s.n.], 2019. <https://youtu.be/-n1x25yu04w>. Acessado em: 25 de janeiro de 2023.
- UNITY: Arquivo de download Unity. [S.l.: s.n.], 2021. <https://unity3d.com/pt/get-unity/download/archive>. Acessado em: 25 de janeiro de 2023.
- UNITY Quaternion: Unity Documentation. [S.l.: s.n.], 2023. <https://docs.unity3d.com/ScriptReference/Quaternion.html>. Acessado em: 25 de janeiro de 2023.