



**POSICIONAMENTO E EXIBIÇÃO DE IMAGENS 3D UTILIZANDO
ÓCULOS DE REALIDADE AUMENTADA PARA APLICAÇÃO
CIRÚRGICA**

Relatório final na modalidade de auxílio à iniciação científica, submetido à Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

Processo: 2020/15835-4

Pesquisador Responsável: Dr. Glauco Augusto de Paula Caurin

Coorientador: Paulo Henrique Polegato

Beneficiário: Calvin Suzuki de Camargo

Informações gerais do projeto

- Título do projeto:

Posicionamento e exibição de imagens 3D utilizando óculos de realidade aumentada para aplicação cirúrgica

- Pesquisador responsável:

Dr. Glauco Augusto de Paula Caurin

- Coorientador:

Paulo Henrique Polegato

- Beneficiário:

Calvin Suzuki de Camargo

- Número do processo do projeto:

2020/15835-4

- Instituição sede do projeto:

Escola de Engenharia de São Carlos da Universidade de São Paulo

- Período de vigência proposto:

01 de março de 2021 a 28 de fevereiro de 2022

- Período coberto por este relatório científico:

01 de março de 2021 a 28 de fevereiro de 2022

Resumo

Trata-se do relatório final do projeto de pesquisa de iniciação científica que comprehende os trabalhos iniciados em março de 2021 até o final de fevereiro de 2022. O projeto visa estudar a exibição de objetos 3D em óculos inteligentes (*Smart glasses*) com realidade aumentada, que podem servir como um dispositivo auxiliar em aplicações cirúrgicas. Para isso, conceituamos as relações entre visão computacional e computação gráfica no campo da realidade aumentada. Com o apoio do Laboratório Aeronáutico de Tecnologias (AeroTech), definimos uma arquitetura de sistema que permite a sobreposição de um modelo 3D na cabeça de um paciente, indicando pontos de implantação de eletrodos durante o procedimento neurocirúrgico orientado por estereoeletroencefalografia (SEEG). Essa arquitetura foi baseada na comunicação de computador e óculos pela rede local: O computador recebe as imagens da câmera dos óculos e utiliza o método de detecção por marcadores fiduciais (*ArUco*) para estimar a posição do paciente e envia as coordenadas aos óculos, indicando onde a projeção deve ser exibida. Os resultados parciais da aplicação apresentam uma rápida responsividade com o método de estimativa escolhido.

Palavras-chaves: *Smart glasses*, Aplicação cirúrgica, Visão computacional, Realidade aumentada.

Sumário

Informações gerais do projeto	i
Resumo	ii
1 Resumo do projeto	1
1.1 Enunciado do problema	1
1.2 Objetivo	2
1.3 Metodologia	2
1.4 Breve histórico do projeto	3
2 Realizações	4
2.1 Calibração	4
2.1.1 Rotação	4
2.1.2 Translação	5
2.1.3 <i>Field of View</i>	5
2.1.4 Distorção	6
2.2 Desenvolvimento Unity	8
2.2.1 Ajuste dos intrínsecos	8
3 Proposta de extensão	10
3.1 Objetivo	11

3.2	Metodologia	11
3.3	Cronograma	13
4	Participação de eventos acadêmicos	14
	Referências bibliográficas	15

1 Resumo do projeto

1.1 Enunciado do problema

Atualmente na medicina, as cirurgias têm o objetivo de causar a mínima agressão ao organismo do paciente, isso implica em um menor tempo de hospitalização; menor incidência de complicações pós-operatórias; menos dor; e recuperação mais rápida (IPC..., 2019). Contudo, entre os diversos desafios de sua realização, a habilidade visual do médico, que envolve a capacidade de localizar e identificar os tecidos sensíveis, é essencial para redução dos danos ao paciente durante um procedimento cirúrgico, nos levando à questão: Como podemos potencializar a visão do médico para aumentar a qualidade dessas operações?



Figura 1.1: Exemplo da utilização de neuronavegador em cirurgia. Fonte: (SERVIÇO..., 2019).

Para esse fim, um *software* que informa e assiste o médico em tempo real na operação denota-se como um ótimo meio de aumentar as chances de sucesso dessas cirurgias. Atualmente, esses *softwares* são muito presentes no planejamento e execução das operações, entretanto, toma-se como exemplo os sistemas de neuronavegação, cuja função é permitir visualização, de forma precisa, das estruturas cerebrais do paciente. O neurocirurgião precisa virar a cabeça para ler os monitores e perde o foco na atividade conduzida, colaborando para um maior tempo de operação e favorecendo a exaustão do profissional, de modo que aumente seu potencial de erros durante os procedimentos cirúrgicos (Figura 1.1).

Para isso, a tecnologia de realidade aumentada (*AR*) oferece o potencial de reduzir es-

sas limitações, em razão de que figuras tridimensionais podem sobrepor a visão do médico para facilitar a visualização e a localização dos objetivos da operação enquanto mantém seus olhos direcionados ao paciente (CHO et al., 2020a). O que foi feito nesse projeto é uma pesquisa que envolve visão computacional e computação gráfica no campo do *AR* e uma aplicação para os óculos da empresa *Seiko Epson Corporation*[®], modelo *Moverio BT-350™*. Recebendo suporte físico e técnico da equipe do Laboratório Aeronáutico de Tecnologias (Aerotech) da Universidade de São Paulo, esse estudo corrobora com o objetivo do grupo que é desenvolver uma plataforma colaborativa de multi-aplicação neurocirúrgica.

1.2 Objetivo

Exibir informações e posicionar modelos tridimensionais em uma região do espaço com *AR*, de forma que facilite o acesso do cirurgião à informação durante a neurocirurgia; estudar e registrar a resposta dos equipamentos utilizados no quesito de qualidade gráfica e latência de resposta do sistema. Tudo isso, com o objetivo central de aumentar a proximidade do cirurgião com a tecnologia de *AR* como apoio durante os procedimentos cirúrgicos.

1.3 Metodologia

Consiste na listagem de possíveis soluções, técnicas ou ferramentas; o estudo e a discussão sobre elas, em seguida, sua implementação. Paralelamente a isso, a busca bibliográfica é constantemente realizada com o objetivo de esclarecer dúvidas sobre os meios imaginados e discutidos com o orientador e coorientador. Essa busca enfatiza os resultados encontrados pelos artigos, o objetivo disto é caracterizar os prós e contras das diversas opções encontradas. Essa pesquisa de artigos trás uma evolução do discernimento dos assuntos do estado da arte, refletindo em uma noção de funcionamento dos métodos científicos que constrói um novo pesquisador no desenvolvimento dessa iniciação científica.

1.4 Breve histórico do projeto

Desde o início dos trabalhos no projeto, foram experimentados diversos tipos de contato com a elaboração de *softwares* para o sistema operacional *Android* (Figura 1.2a); testes das ferramentas da documentação dos óculos de realidade aumentada *Moverio BT-350* (Figuras 1.2b e 1.2c); e a elaboração de aplicativos que ilustram o objetivo do *VCranium* (Figura 1.2d). Assim, como foi explicado no relatório parcial da pesquisa, pretendíamos prosseguir o desenvolvimento estabelecendo uma arquitetura composta por computador e óculos para capturar os dados necessários para a projeção em realidade aumentada, os detalhes serão descritos no capítulo das realizações.



Figura 1.2: Histórico de realizações da pesquisa até a primeira entrega parcial. Fonte: Autor.

2 Realizações

2.1 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: Realizado com o *RVEC* de *Rodrigues* de cada *frame*;

Translação: Realizado com o *TVEC* de *Rodrigues* 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;

2.1.1 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:

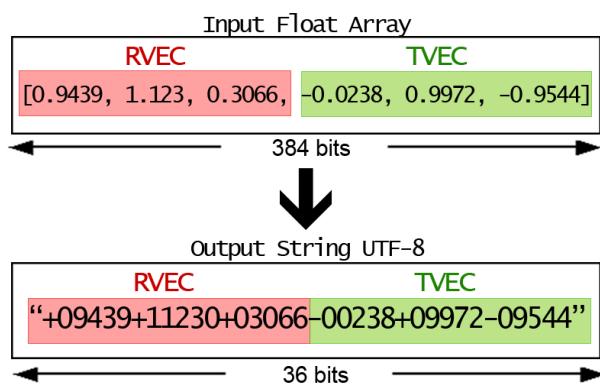


Figura 2.1: Estrutura da mensagem via *socket*. Fonte: Autor.

Um método foi implementado para a 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 2.1. 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.

2.1.2 Translação

Da mesma forma que a rotação, a translação é recebida do *TVEC* da mensagem da Figura 2.1. 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 2.2.



Figura 2.2: Exemplo de sobreposição sem calibração intrínseca. Fonte: Autor.

2.1.3 *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* [FONTE]:



Figura 2.3: 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.

2.1.4 Distorção

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

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.

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

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

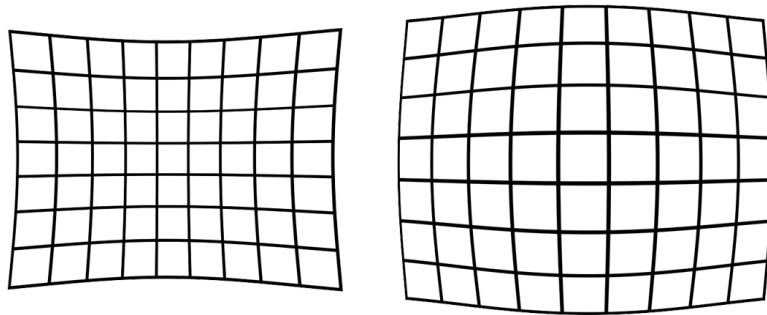


Figura 2.4: Exemplos de distorção radial. Fonte: Autor.

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.

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

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

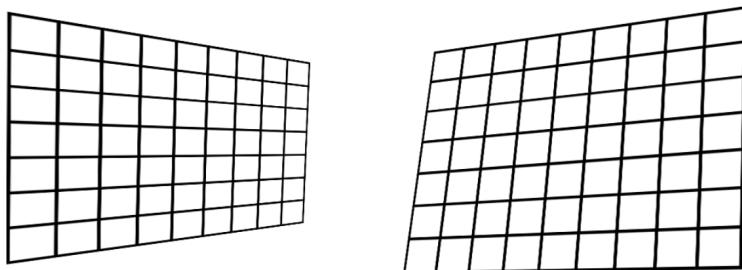


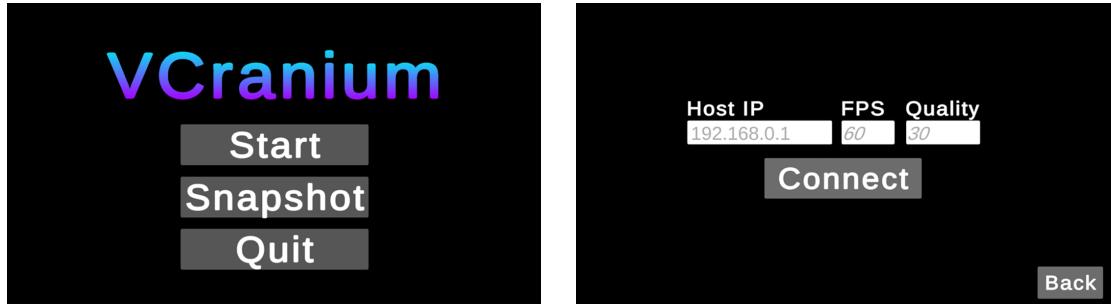
Figura 2.5: Exemplos de distorção tangencial. Fonte: Autor.

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*:

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

2.2 Desenvolvimento Unity

Trazendo a teoria apresentada na Seção 2.1 para a realidade no ambiente de desenvolvimento do *Unity*. Para isso, deve-se organizar de "cenas", onde o usuário pode transitar utilizando a interface, como na Figura 2.6, 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 2.6: Imagens da interface do *VCraniun*. Fonte: Autor.

2.2.1 Ajuste dos intrínsecos

A Seção 2.1 também explicita que obtemos todos os parâmetros intrínsecos e de distorção de uma câmera após a calibração com o método do *chessboard* (Figura 2.3). Porém, é necessário um método que realize a calibração da câmera em que o *VCraniun* 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.

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).

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.

Certamente a Figura 2.2 mostra que ao fim da calibração extrínseca, a sobreposição é visualmente boa, i.e., em boa parte da ilustração, o erro da sobreposição é insignificante e, com isso, não apresenta problemas para demonstração de conceitos.

3 Proposta de extensão

O projeto até o momento enfrentou diversos desafios e buscou desempenhar muitos testes até decidirmos escolher um meio de começar a elaboração do *VCranium* (Capítulo "??"). A pesquisa bibliográfica mostrou a possibilidade, por meio de diferentes técnicas, de aplicarmos o *Moverio BT-350* no ambiente cirúrgico e, especialmente, na neurocirurgia (CHO et al., 2020b). A extensão do período de trabalho no projeto, assim como a renovação da bolsa, trará também o aprofundamento e refinamento dos processos implementados no programa.

Com a arquitetura do sistema montada, temos um cenário propício para continuar a desenvolver atualizações que permitam uma melhor estabilidade e compatibilidade com os recursos dos óculos. Isto posto, podemos não só refinar o algoritmo atual de detecção por marcadores, como podemos aplicar redes treinadas de *machine learning* ou utilizar o modelo da *Intel RealSense* do laboratório para o uso de *surface matching* da superfície da cabeça do paciente, ambas técnicas já sendo estudadas por outros pesquisadores do *AeroTech*.

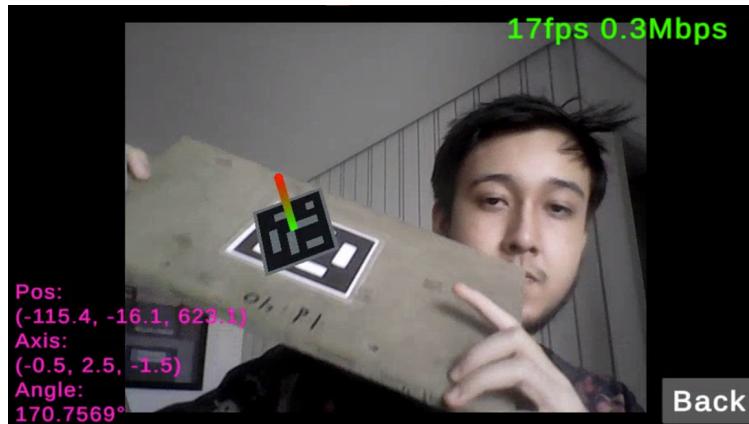


Figura 3.1: Erro de projeção por falta de calibração da câmera. Fonte: Autor.

O próximo passo do *VCranium* é a calibração da projeção em realidade aumentada. Visto que a observação do usuário dos óculos é estereoscópica, a primeira parte de viabilizar a exibição de AR é fazer a visualização independente para cada olho. Para isso, é necessário um cálculo de calibração particular de cada usuário, e além disso, garantir que essa calibração se mantenha em todas as posições da projeção no espaço virtual. Atualmente, o projeto se encontra desregulado na projeção, não criando o efeito de sobreposição adequado (Figura 3.1).

3.1 Objetivo

Trabalhar na arquitetura de sistema montado no primeiro período da pesquisa, que envolverá os compromissos de refinar o método atual de estimativa de posição por marcadores; experimentar diferentes métodos de visão computacional; e comparar com os dados de precisão da literatura. Dessa maneira, cooperando com o objetivo primordial de aumentar a proximidade do neurocirurgião com a tecnologia *AR* em procedimentos cirúrgicos.

3.2 Metodologia

Como trata-se de um projeto com um sistema já definido, a metodologia precisa dar ênfase no conhecimento da confiabilidade dos métodos e, por isso, consistirá no estudo da matemática das projeções da computação gráfica. Esse assunto é muito importante para entendermos como funciona a visualização 3D de objetos em um espaço virtual, os cálculos são feitos com transformações por meio da multiplicação de matrizes (*Model*, *View* e *Projection*), que por sua vez, agem como parâmetros desse método (LEARN..., 2022).

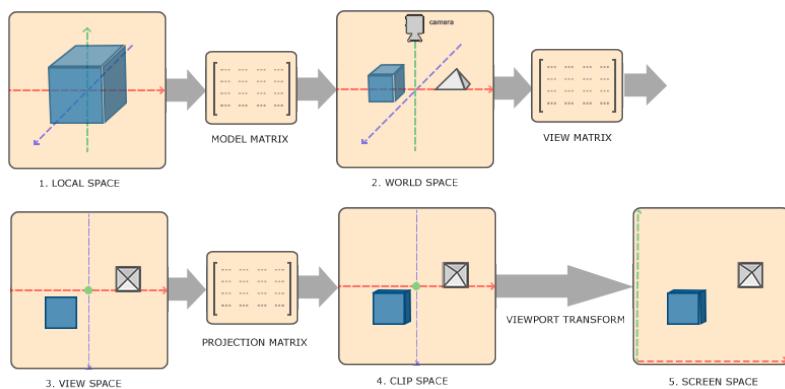


Figura 3.2: Aplicação das matrizes *Model*, *View* e *Projection* para a visualização 3D em perspectiva.
Fonte: (LEARN..., 2022).

Não se limitando apenas na teoria, mas também estudando a aplicação da computação gráfica em realidade aumentada, podemos então criar uma relação entre as visualizações 3D do espaço virtual dos óculos e a tela em que serão projetadas as imagens, sendo essa, a base do efeito de sobreposição em *AR*. Portanto, esse trabalho vai enfatizar os estudos de marcadores fiduciais para essa visualização, sendo necessário correlacionar os parâmetros da câmera com os modelos

com o intuito de fazer uma projeção de forma adequada (TENG; CHEN, 2012).

Por fim, o trabalho exigirá uma aferição da precisão da projeção em realidade aumentada apresentada pelo método. Na literatura, essa medida foi adquirida com a utilização de ferramentas que entregam uma estimativa do erro da projeção (Figura 3.3) (MARUYAMA et al., 2018). Ainda que a metodologia terá destaque no aprofundamento teórico dos métodos, as pesquisas de novos artigos serão feitas conjuntamente com esses estudos, procurando monitorar as descobertas e novos resultados no campo de realidade aumentada aplicada em cirurgias e neurocirurgias.

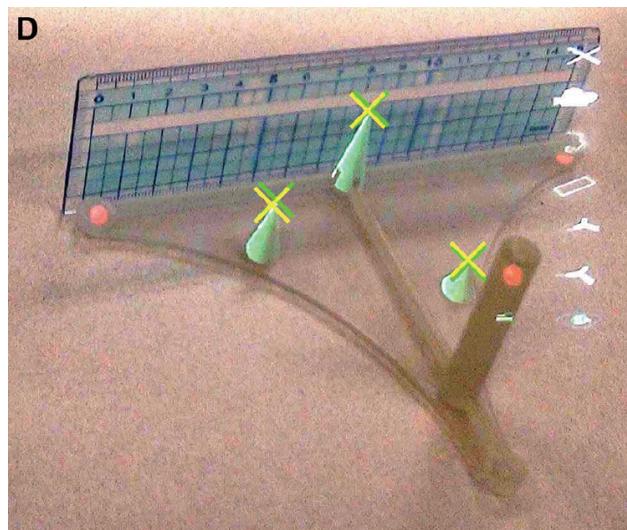


Figura 3.3: Aplicação do *phantom* para a medição do erro de projeção. Fonte: (MARUYAMA et al., 2018).

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	6
Revisão literária de aplicação de marcadores em AR							
Estudo das transformações matriciais de perspectiva							
Estudo sobre calibração de projeção AR							
Aprofundamento em calibração estereoscópica Moverio							
Testes com outros algoritmos de detecção							
Coleta de dados da precisão de métodos*							

Tabela 3.1: *A coleta de dados vai envolver o auxílio 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 detectar a posição de marcadores em tempo real e posicionar um modelo virtual do *Unity* (aproximadamente 15 ciclos por segundo). Porém, utilizar somente um marcador traz diversas limitações de movimento ao usuário dos óculos, portanto, é necessário uma adaptação do sistema para a ampliar seu alcance e superar os problemas de oclusão do marcador.

Referências bibliográficas

- 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.
- CHO, J. et al. Enhancing Reality: A Systematic Review of Augmented Reality in Neuronavigation and Education. **World Neurosurgery**, 2020. ISSN 18788750. DOI: 10.1016/j.wneu.2020.04.043.
- IPC Instituto de Patologia da Coluna. [S.l.: s.n.], 2019. <https://neurocirurgia.com/cirurgia-minimamente-invasiva/>. Acessado em: 18 de março de 2022.
- LEARN OpenGL: Coordinate Systems. [S.l.: s.n.], 2022. <https://learnopengl.com/Getting-started/Coordinate-Systems>. Acessado em: 21 de março de 2022.
- MARUYAMA, K. et al. Smart Glasses for Neurosurgical Navigation by Augmented Reality. **Operative Neurosurgery**, v. 15, n. 5, p. 551–556, nov. 2018. ISSN 23324260. DOI: 10.1093/ons/oxz279. Disponível em: <<https://academic.oup.com/ons/article/15/5/551/4823484>>.
- SERVIÇO de Neurocirurgia retira lesão cerebral com paciente acordado: Youtube. [S.l.: s.n.], 2019. <https://youtu.be/-n1x25yu04w>. Acessado em: 5 de março de 2022.
- TENG, C. H.; CHEN, J. Y. An augmented reality environment for learning OpenGL programming. **Proceedings - IEEE 9th International Conference on Ubiquitous Intelligence and Computing and IEEE 9th International Conference on Autonomic and Trusted Computing, UIC-ATC 2012**, p. 996–1001, mai. 2012. DOI: 10.1109/UIC-ATC.2012.57.
- UNITY: Arquivo de download Unity. [S.l.: s.n.], 2021. <https://unity3d.com/pt/get-unity/download/archive>. Acessado em: 10 de agosto de 2021.