



CENTRO DE MATEMÁTICA, COMPUTAÇÃO E COGNIÇÃO  
PROJETO DE GRADUAÇÃO EM BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

VITTÓRIA ARIEL DOS SANTOS BOROTTO  
Orientador: Prof. Mario Gazziro

**ANÁLISE COMPARATIVA DA EFICIÊNCIA DE SENSORES RGB-D  
PARA ESTIMAÇÃO DE POSES ANTROPOMÉTRICAS**

Santo André, SP  
2024

UNIVERSIDADE FEDERAL DO ABC  
CENTRO DE MATEMÁTICA, COMPUTAÇÃO E COGNIÇÃO  
PROJETO DE GRADUAÇÃO EM BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

VITTÓRIA ARIEL DOS SANTOS BOROTTO

**ANÁLISE COMPARATIVA DA EFICIÊNCIA DE SENSORES RGB-D PARA  
ESTIMAÇÃO DE POSES ANTROPOMÉTRICAS**

Monografia apresentada ao Curso de Bacharelado em  
Ciência da Computação da Universidade Federal do  
ABC como parte dos requisitos necessários para a  
obtenção do grau de Cientista da Computação.

**Orientador:** Prof. Mario Gazziro

Santo André, SP  
2024



## Ata nº 8/2024/CMCC

Na data de 02 de Maio de 2024, às 16h00, na Universidade Federal do ABC em Santo André foi realizada a defesa de trabalho intitulado:

Análise comparativa da eficiência de sensores rgdb para estimação de poses antropométricas, para conclusão do Projeto de Graduação em Computação III do bacharelado em Ciência da Computação da discente Vittoria Ariel dos Santos Borotto, RA11201811288. Esteve presente avaliando o trabalho a banca examinadora composta pelos professores: Mario Alexandre Gazziro (UFABC), Hugo Puertas de Araújo (UFABC), Eduardo Machado Real (UEMS). Após apresentação do trabalho pela discente e arquição da banca examinadora, seu trabalho foi considerado:

Aprovado com conceito A

Aprovado com conceito \_\_\_\_\_, sob a condição de apresentação de revisões sugeridas pela banca até o dia \_\_\_\_/\_\_\_\_/\_\_\_\_.

Reprovado

Santo André, 02 de Maio de 2024

\_\_\_\_\_  
Mario Alexandre Gazziro  
orientador(a)

\_\_\_\_\_  
Hugo Puertas de Araújo  
examinador(a) titular

\_\_\_\_\_  
Eduardo Machado Real  
examinador(a) titular

\_\_\_\_\_  
Victoria Alejandra Salazar Herrera  
examinador(a) suplente

\_\_\_\_\_  
Filipe Ieda Fazanaro  
examinador(a) suplente

*Dedico este trabalho às minhas famílias, sejam elas as de sangue ou as adquiridas ao longo de nossas vidas. Seu constante apoio tornou esta jornada possível.*

# Agradecimentos

Primeiramente gratidão a Deus, pois sem a minha fé, não teria sido possível concluir esta jornada acadêmica. Os meus sinceros agradecimentos aos meus pais, Cassia e Valter, por investirem em minha educação e apoiarem meus sonhos desde o início da faculdade, inclusive nos sonhos que surgiram ao longo da jornada acadêmica. Obrigada por tornarem esta jornada possível.

Expresso profundos agradecimentos à minha avó Sandra e ao meu irmão Pedro por celebrarem sempre minhas conquistas e por fazerem parte ativa da minha vida, oferecendo cuidado e apoio ao longo desses anos. Sua presença constante não apenas fortaleceu meu percurso acadêmico, mas também contribuiu significativamente para a construção da pessoa que sou. Minha sincera admiração a vocês.

Agradeço imensamente às minhas amigas e irmãs que a UFABC me proporcionou: Rebeca, Le Rossato, Le Bonassa. Mesmo trilhando caminhos distintos, vocês foram calma, suporte e apoio fundamentais para a conclusão deste capítulo acadêmico. Obrigada por tudo. Expresso também minha gratidão a amigos essenciais que a faculdade me presenteou: David Pianura, Felipe Moreira, Lucas Medeiros. Agradeço pelo suporte constante e companheirismo ao longo desta formação.

Agradeço imensamente às minhas amigas de longa data, Giovana Aguiar, Lais Alberto, Gabriela Farias, Luisa Pataquini, Giulia Menegasso, Denise Castilho e Stephany Phillipetti por me apoiarem desde antes mesmo do início desta jornada acadêmica e por me suportarem ao longo dessa caminhada. Juntas, testemunhamos o nascimento e a concretização desses sonhos. Obrigada por tudo.

E por fim agradeço ao Professor Mario Gazziro pela paciência e pela oportunidade de ser orientada por um profissional tão competente, e ao Professor Jecel Mattos de Assumpção Jr que contribuiu com a realização e finalização desse projeto. Expresso também minha gratidão à Universidade Federal do ABC por contribuir significativamente para a construção do profissional que sou e pelo conhecimento que adquiri ao longo desse percurso acadêmico.

“A coragem é encontrada em lugares improváveis”. (Tolkien)

# Resumo

Este projeto explora o avanço dos gêmeos digitais através da lente da tecnologia de esqueletização de movimentos corporais, crucial para a criação de avatares digitais controláveis e interativos. Originados na década de 1970, os gêmeos digitais têm evoluído de simples réplicas virtuais para sistemas complexos que permitem interações dinâmicas entre os mundos físico e virtual. Acompanhando essa evolução, a tecnologia de captura de movimento tem se destacado como um componente essencial para a precisão e eficácia dos gêmeos digitais, especialmente em aplicações que demandam interações naturais e intuitivas. Contudo, o custo e a capacidade técnica de tais tecnologias variam significativamente, levantando questões sobre a viabilidade econômica e a otimização tecnológica para aplicações práticas.

Diante desse contexto, nosso estudo visa avaliar e comparar a eficiência e qualidade da esqueletização de movimentos corporais para controle de avatares digitais, utilizando duas tecnologias distintas: Kinect e NuiTrack, em combinação com o sensor Astral S. Os objetivos específicos incluem: determinar a qualidade da esqueletização proporcionada pelo NuiTrack, considerando a precisão dos pontos de articulação, fluidez dos movimentos e latência; comparar as capacidades de body tracking entre Kinect e NuiTrack (com o sensor Astral S) em termos de precisão, latência e desempenho geral; aplicar a análise de distância de Chamfer para avaliar a concordância entre os sistemas e, finalmente, determinar qual dos sistemas oferece a maior eficácia para a captura e reprodução de movimentos corporais.

Através desta análise comparativa, o projeto busca não apenas avançar nossa compreensão sobre a otimização de tecnologias de captura de movimento para gêmeos digitais, mas também contribuir para a discussão sobre como equilibrar inovação tecnológica com acessibilidade financeira. Porém, os resultados obtidos não corroboraram a hipótese de similaridade entre os sensores para esse tipo de aplicação, determinação de pose para uso em gêmeos digitais.

**Palavras-chave:** Esqueletização de Movimentos. Captura de Movimento. Gêmeo Digital. Kinect. NuiTrack. Controle de Avatares Digitais. Tecnologias de Body Tracking. Interatividade Virtual. Sensor Astral S. Body Tracking.

# Abstract

This project investigates the advancement of digital twins through the lens of body movement skeletonization technology, crucial for the creation of controllable and interactive digital avatars. Originating in the 1970s, digital twins have evolved from simple virtual replicas to complex systems enabling dynamic interactions between the physical and virtual worlds. Alongside this evolution, motion capture technology has emerged as a key component for the accuracy and effectiveness of digital twins, especially in applications requiring natural and intuitive interactions. However, the significant variance in the cost and technical capability of such technologies raises questions about their economic viability and technological optimization for practical applications.

Against this backdrop, our study aims to evaluate and compare the efficiency and quality of body movement skeletonization for controlling digital avatars, utilizing two distinct technologies: Kinect and NuiTrack, in combination with the Astral S sensor. Specific objectives include: assessing the quality of skeletonization provided by NuiTrack, considering the accuracy of joint points, movement fluidity, and system latency; comparing the body tracking capabilities between Kinect and NuiTrack (with the Astral S sensor) in terms of precision, latency, and overall performance; applying Chamfer distance analysis to assess the concordance between the systems; and ultimately determining which system offers greater efficacy in capturing and reproducing body movements.

Through this comparative analysis, the project seeks not only to advance our understanding of optimizing motion capture technologies for digital twins but also to contribute to the discussion on how to balance technological innovation with financial accessibility. However, the results obtained did not corroborate the hypothesis of similarity between sensors for this type of application, pose determination for use in digital twins.

**Keywords:** Movement Skeletonization. Motion Capture Technology. Digital Twin. Kinect. NuiTrack. Digital Avatar Control. Body Tracking Technologies. Virtual Interactivity. Sensor Astral S. Body Tracking.



# Lista de Ilustrações

Figura 2.1 – Triangulação - fotogrametria . . . . .	8
Figura 2.2 – Processo de obtenção de malha 3D . . . . .	10
Figura 2.3 – Distância de Chamfer . . . . .	12
Figura 3.1 – Diagrama de Configuração dos Componentes . . . . .	14
Figura 3.2 – Geometria da técnica de triangulação: as lentes são localizadas na origem, com distância focal $f$ para o plano da imagem e a linha base $b$ entre o projetor e a câmera. . . . .	15
Figura 3.3 – Sensor ASTRA STEREO S U3 . . . . .	16
Figura 3.4 – Microsoft Kinect . . . . .	17
Figura 3.5 – Placa de vídeo Titan XP . . . . .	18
Figura 3.6 – Placa de vídeo Rtx 3060 . . . . .	18
Figura 3.7 – Plataforma giratória utilizada na rotação do manequim . . . . .	19
Figura 3.8 – Software NuiTrack capturando imagens do manequim. . . . .	20
Figura 3.9 – Software NuiTrack . . . . .	21
Figura 3.10–Manequim em plataforma giratória. . . . .	22
Figura 3.11–Tabela de Setup A e B . . . . .	22
Figura 3.12–Configuração Sensores para Captura . . . . .	23
Figura 3.13–Diagrama de captura de dados. . . . .	23
Figura 4.1 – Resultado final da comparação entre os sensores . . . . .	24

# Lista de Abreviaturas e Siglas

2D	Bidimensional
3D	Tridimensional
CCD	Charged-coupled Device
Kd	Ganho derivativo
Ki	Ganho integral
Kp	Ganho proporcional
LIDAR	Light Detection And Ranging
PID	Proporcional, Integral, Derivativo
PWM	Pulse Width Modulation
RGB	Red, Blue, Green
RGB-D	Red Green Blue-Depth

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Justificativa	1
1.2	Contextualização	3
1.3	Objetivos	4
1.3.0.1	Objetivo Principal	4
1.3.0.2	Objetivos Específicos	4
<b>2</b>	<b>Revisão Bibliográfica</b>	<b>5</b>
2.1	Fundamentação Teórica	5
2.1.1	Poses Antropométricas	5
2.2	Body tracking	5
2.2.1	Conceitos de Avatares e Gêmeos Digitais	6
2.2.1.1	Gêmeos Digitais	6
2.2.1.2	Avatares	6
2.2.1.3	Interrelação entre Gêmeos Digitais e Avatares	7
2.2.2	Mimetismo Virtual	7
2.2.3	Triangulação	8
2.2.4	Nuvem de pontos	9
2.2.5	Malha	9
2.2.6	Nuitrack	10
2.2.7	Distância de Chamfer	12
2.3	Trabalhos Relacionados	12
<b>3</b>	<b>Metodologia</b>	<b>14</b>
3.1	Componentes	14
3.1.1	Sensores RGB-D	14
3.1.1.1	Estimação de distância em sensores RGB-D	15
3.1.1.2	Sensor Orbbec ASTRA STEREO S U3	16
3.1.1.3	Kinect	17
3.1.2	Placa de vídeo	17
3.1.3	Plataforma giratória	18
3.1.3.1	Controle de Velocidade	18
3.1.4	Nuitrack	20
3.1.5	Manequim	20
3.2	Preparação e Configuração do Experimento	20
3.3	Captura de Poses Antropométrica	22
<b>4</b>	<b>Resultados</b>	<b>24</b>
<b>5</b>	<b>Conclusão</b>	<b>25</b>

<b>Referências</b> . . . . .	<b>26</b>
<b>A Apêndices - Código fonte</b> . . . . .	<b>28</b>
A.0.1 Nitrack SDK Script . . . . .	28
A.0.2 Chamfer Script . . . . .	32

# 1 Introdução

## 1.1 Justificativa

O conceito de *digital twin* (do inglês, gêmeo digital) foi pioneiramente implementado pela primeira vez pela Administração Nacional da Aeronáutica e Espaço (NASA), no programa Apollo 13, evoluiu significativamente desde suas origens. Com o objetivo de permitir a comunicação e o controle durante a missão espacial, onde foi criado um veículo idêntico ao que realizou a missão no espaço para ser monitorado na Terra. Porém, quem primeiro cunhou o termo foi o professor da Universidade de Michigan, nos EUA, Michael Grieves, em 2003 [Garcia Bianca Gabalde \(2022\)](#).

Na primeira definição de Grieves, o gêmeo digital foi definido como componentes físicos, produtos ou sistemas em um espaço físico, sua correspondente representação virtual em um espaço virtual e o fluxo de informação entre estes dois espaços [Garcia Bianca Gabalde \(2022\)](#).

O Digital Twin Consortium, iniciativa global para a aceleração do desenvolvimento, adoção, interoperabilidade e segurança de gêmeos digitais, define gêmeo digital da seguinte maneira: um gêmeo digital é uma representação virtual de ativos ou processos do mundo real, sincronizados a uma frequência (de aquisição de dados) e fidelidade (dos dados) especificadas. O gêmeo digital usa dados históricos e em tempo real para representar respectivamente o passado e o presente e prever e simular o futuro. O gêmeo digital é potencializado pela integração, centrado em dados e implementado em sistemas de IT/OT ou Tecnologia de Informação e Tecnologia Operacional, em português [Garcia Bianca Gabalde \(2022\)](#).

A empresa de pesquisas Gartner, em 2018, previu que os gêmeos digitais seriam uma das 10 (dez) maiores tendências tecnológicas nos próximos dez anos [Garcia Bianca Gabalde \(2022\)](#). Embora o uso desse tipo de tecnologia já venha sendo difundido em inúmeras áreas tais como: manufatura, aviação, aeroespacial e saúde, ele ainda é limitado à empresas de grande porte. Nessas empresas, o gêmeo virtual dos sistemas físicos é utilizada para fazer previsões sobre condições adversas de funcionamento, testes de colisão e de aerodinâmica, para citar alguns exemplos de aplicações. Além disso, essa tecnologia também vem sendo implementada por companhias cinematográficas e de efeitos visuais com auxílio de câmeras e sensores do tipo LiDAR ToF, que possibilitam a reconstrução de forma quase instantânea e com alta precisão do ambiente e objetos.

Embora com um grande potencial mercadológico – globalmente de US\$3 bilhões em 2018, com perspectiva de crescimento de 28% ao ano entre 2019-2025, atingindo mais de US\$20 bilhões em 2025 ([D’MELLO, 2019](#)) - a sua implementação pelas empresas vem apresentando um lento crescimento. Pesquisa realizada pela Gartner em 2019, indicou que apenas 13% das organizações que já implementaram projetos de IoT, usavam também gêmeos digitais, enquanto

62% estavam ainda no processo de estabelecimento do uso do gêmeo digital ou no plano para fazê-lo (COSTELLO, 2019).

De maneira simplificada o termo digital twin é definido como a réplica virtual de um produto, sistema ou ser vivo que são continuamente atualizados com dados de sua contraparte física, bem como do ambiente no qual está inserido. Ele é o elo entre o mundo virtual e as entidades físicas e, por isso é considerado um dos pilares da Indústria 4.0 e, segundo Jiang et al. (2021), é a força motriz para futuras inovações.

A relevância dos gêmeos digitais, conforme evidenciado por previsões de crescimento de mercado e adoção crescente em setores estratégicos, ilustra não apenas seu valor econômico, mas também sua capacidade de impulsionar inovações. No entanto, a implementação eficaz de gêmeos digitais exige uma representação precisa e em tempo real dos sistemas físicos, um desafio particularmente pronunciado quando aplicado ao mapeamento e controle de movimentos humanos para avatares digitais. E vem se tornando cada vez mais popular e procurado pela indústria 4.0. Acentuado com o surgimento das plataformas de metaverso se fazem necessários o surgimento de equipamentos e tecnologia capazes de digitalizar pessoas e objetos para criação de seus respectivos avatares na forma de gêmeos digitais para uso em marketplaces e ambientes de entretenimento e interação social dentro do metaverso.

Neste contexto, a justificativa para o foco do presente trabalho da "análise comparativa de eficiência de sensores RGB-D para estimação de poses antropométricas" reside na identificação de uma lacuna crítica: a necessidade de comparar e avaliar a eficácia de diferentes tecnologias de sensores para body tracking. Essa comparação é essencial, dado que a precisão na captura de movimento é fundamental para a criação de gêmeos digitais que possam ser controlados em tempo real, permitindo aplicações que vão desde a reabilitação física até a criação de experiências imersivas em realidade virtual para entretenimento.

Ademais, a introdução de novas tecnologias de sensores, como as baseadas em LiDAR ToF, promete melhorias significativas na precisão e na velocidade de captura de movimentos, mas a sua aplicabilidade prática e eficácia relativa em comparação com tecnologias existentes permanecem questões em aberto.

A escolha entre tecnologias de captura de movimento de alto custo e alternativas mais acessíveis representa um dilema crucial para a implementação efetiva de gêmeos digitais para estimação de poses antropométricas. Este projeto propõe uma análise comparativa entre um sensor de alto custo e um de custo reduzido, como o Kinect, visando não apenas avaliar a disparidade em termos de performance e precisão, mas também entender como o custo influencia a viabilidade e a adoção dessas tecnologias. A relevância desta comparação se acentua no contexto atual, onde a acessibilidade financeira pode ser tanto um facilitador quanto um obstáculo para a inovação e aplicação prática dos gêmeos digitais.

Portanto, a justificativa para este trabalho encontra-se na interseção de uma oportunidade

tecnológica emergente e uma necessidade prática de avançar na compreensão de sensores RGB-D para estimação de poses antropométricas no mapeamento de movimentos humanos. Ao focar na comparação de tecnologias sensoriais para body tracking, este projeto visa não apenas contribuir academicamente para o campo, mas também oferecer direções práticas para a implementação efetiva de avatares digitais controlados, potencializando aplicações em diversas áreas.

## 1.2 Contextualização

Os gêmeos digitais surgiram como um conceito revolucionário na década de 1970, inicialmente desenvolvido pela NASA para criar simulações precisas de veículos espaciais que permitissem um monitoramento e controle mais eficazes em missões espaciais. Essa inovação marcou o início de uma era de exploração digital, expandindo-se para além dos confins da aeroespacial até alcançar uma gama diversificada de setores, incluindo manufatura, saúde e entretenimento. Ao longo dos anos, à medida que a tecnologia avançou, os gêmeos digitais evoluíram de simples réplicas virtuais para sistemas complexos que interagem dinamicamente com seus correspondentes físicos, impulsionados por dados em tempo real.

Este desenvolvimento foi paralelamente acompanhado pelo avanço das tecnologias de sensoriamento e captura de movimento, cruciais para a precisão e eficácia dos gêmeos digitais. A capacidade de capturar e replicar movimentos humanos com alta fidelidade é fundamental, especialmente em aplicações que requerem interações naturais e intuitivas entre humanos e ambientes virtuais. Essa necessidade transcende o entretenimento, alcançando a reabilitação física, a simulação de procedimentos médicos, o treinamento esportivo, e até mesmo o design de espaços de trabalho ergonômicos.

Nesse contexto, a tecnologia de captura de movimento se destaca como um pilar para o aprimoramento de gêmeos digitais, particularmente no que tange à criação de avatares digitais controláveis. Os avanços recentes nesta área, incluindo o desenvolvimento de sensores baseados em LiDAR ToF, oferecem possibilidades promissoras para melhorias na precisão, velocidade e adaptabilidade da captura de movimentos. Contudo, a implementação dessas tecnologias avançadas frequentemente enfrenta um obstáculo significativo: o custo. O aumento dos preços torna algumas das soluções mais sofisticadas financeiramente inacessíveis para muitos projetos e pesquisas. Em contrapartida, alternativas de menor custo, como o Kinect, embora possam apresentar limitações em termos de precisão e capacidade, surgem como soluções viáveis, equilibrando a relação custo-benefício.

Essa dicotomia entre custo e capacidade tecnológica estabelece um panorama complexo e desafiador, no qual o presente trabalho se insere. A análise comparativa da eficiência de sensores RGB-D para estimação de poses antropométricas, levando em consideração tanto sensores de alta gama quanto soluções mais acessíveis como o Kinect, destaca-se como um enfoque crítico. Este enfoque é necessário para avançar nossa compreensão sobre como a tecnologia

pode ser otimizada para atender às necessidades de precisão e interatividade, sem descuidar da acessibilidade financeira. Assim, a contextualização serve não apenas para traçar a evolução histórica e as capacidades atuais dos gêmeos digitais e das tecnologias de captura de movimento, mas também para enfatizar a importância de encontrar um equilíbrio entre inovação tecnológica e viabilidade econômica.

## 1.3 Objetivos

### 1.3.0.1 Objetivo Principal

Avaliar e comparar a eficiência e qualidade de sensores RGB-D para estimação de poses antropométricas que são utilizados como base para captura de movimentos corporais para controle de avatares digitais, utilizando as tecnologias Kinect e o sensor Astral S em combinação com o software NuiTrack.

### 1.3.0.2 Objetivos Específicos

1. Avaliar a Qualidade do body tracking com NuiTrack: Determinar a qualidade da esqueletização de movimentos corporais gerada pelo NuiTrack, considerando aspectos como precisão dos pontos de articulação, e a latência na resposta do sistema.
2. Comparar a Eficiência do Body Tracking entre Kinect e Sensor Astral S: Comparar diretamente as capacidades de body tracking do Kinect e do NuiTrack (este último utilizando o sensor Astral S) para identificar qual sistema oferece maior precisão, menor latência e melhor desempenho global na captura de poses antropométricas.
3. Utilizar a Análise de distancia Chamfer para Comparação dos Métodos: Aplicar a análise de distancia Chamfer para avaliar a concordância entre os dados capturados pelos sistemas Kinect e Sensor Astral S com NuiTrack, permitindo uma comparação objetiva da eficiência entre os dois métodos.
4. Determinar o sensor RGB-D mais eficiente para estimação de poses antropométrica: Concluir qual dos dois sistemas, Kinect ou Sensor Astral S, apresenta maior eficácia na captura e reprodução de movimentos corporais para o controle de avatares digitais, com base nos critérios estabelecidos de precisão, e resposta do sistema.



## 2 Revisão Bibliográfica

### 2.1 Fundamentação Teórica

#### 2.1.1 Poses Antropométricas

Poses antropométricas referem-se a configurações específicas do corpo humano que são usadas para medir e analisar as dimensões corporais e as relações espaciais entre diferentes partes do corpo. Essas poses são fundamentais em diversos campos, como ergonomia, design de vestuário, medicina e ciências do esporte, pois fornecem informações cruciais sobre a forma e a função do corpo humano. São usadas para padronizar a postura do corpo durante as medições, garantindo que as medidas sejam consistentes, confiáveis e comparáveis.

Com o avanço tecnológico, as poses antropométricas são frequentemente capturadas e analisadas usando equipamentos de alta precisão, como sensores de movimento, escaneamento 3D e sistemas de captura de movimento, que fornecem dados detalhados sobre a postura e o movimento do corpo.

### 2.2 Body tracking

O Body Tracking, ou rastreamento de corpo, é uma tecnologia que envolve o acompanhamento e mapeamento dos movimentos do corpo humano em tempo real. Essa técnica utiliza sensores e algoritmos avançados para capturar e interpretar os movimentos, posturas e posições das diferentes partes do corpo.

Pode ser realizado com diferentes tipos de sensores, como câmeras RGB-D, câmeras de profundidade, sensores de infravermelho ou até mesmo tecnologias mais avançadas, como LiDAR. Esses sensores capturam informações sobre a posição tridimensional das partes do corpo.

Uma das características essenciais do body tracking é a capacidade de realizar o rastreamento em tempo real. Isso significa que os movimentos do corpo são capturados e refletidos instantaneamente em um ambiente virtual ou aplicação interativa. Pode ter diversas aplicações em setores de entretenimento, como jogos de realidade virtual, simulações interativas e experiências imersivas. Ele permite que os usuários controlem personagens virtuais, realizem gestos específicos e interajam de forma natural com ambientes virtuais.

Na área de saúde e treinamento, o body tracking é utilizado para monitorar e corrigir movimentos em terapias físicas, reabilitação e atividades esportivas. Também é aplicado em treinamentos específicos, como aprimoramento da técnica em esportes.

É fundamental para o desenvolvimento de interfaces de usuário inovadoras e intuitivas.

Ele possibilita interações naturais, como gestos e movimentos, substituindo os tradicionais dispositivos de entrada, como teclados e mouses.

Em resumo, o body tracking desempenha um papel crucial na criação de experiências interativas e imersivas, proporcionando uma forma natural e intuitiva de interação entre o mundo físico e o virtual. Suas aplicações abrangem diversos setores, desde entretenimento até saúde e treinamento.

## **2.2.1 Conceitos de Avatares e Gêmeos Digitais**

### **2.2.1.1 Gêmeos Digitais**

Hoje em dia, um Digital Twin é definido como uma representação virtual de um produto físico, ativo, processo ou sistema em um CPS/CPPS, e ao longo de seu ciclo de vida, capaz de espelhar em tempo real suas características estáticas e dinâmicas como resultado de um conjunto de dados contínuo transmitido entre sua réplica digital e entidade física (Saddik, 2018; Ashtari et al., 2019). Além disso, um gêmeo digital é dividido em três componentes principais: (i) um produto físico no mundo real incorporado com sensores, atuadores, processadores, software e conectividade, também conhecido como “smart Connected product”, (ii) um produto virtual no mundo cibernético representado por um modelo virtual 3D projetado para simulações avançadas e análise de dados que sincroniza o mundo cibernético e o mundo real sistemas, e (iii) a conexão bidirecional de fluxos de dados entre os mundos físico e cibernético que unifica o mundo virtual e produtos reais – também chamados de “digital thread”.

Os Gêmeos Digitais prometem benefícios significativos para suas diferentes partes interessadas quando usado para apoiar o projeto, fabricação, gerenciamento, monitoramento e controle, bem como otimização de produtos manufaturados e equipamentos de produção e sistemas na fabricação. Além disso, o suporte dos Digital Twins ao desenvolvimento de sistemas industriais de produtos-serviços, modelos de negócios que permitem novas possibilidades para o fornecimento de serviços de valor agregado e atualizações físicas em todas as fases do ciclo de vida de um produto, ativo, processo ou sistema (por exemplo, certificação, manutenção, monitoramento, atualização, previsões).

### **2.2.1.2 Avatares**

Um Avatar é definido como uma representação gráfica específica das partes interessadas e uma visualização de informações relacionadas ao produto ao longo do seu ciclo de vida no mundo virtual, estabelecendo interfaces adequadas que acomodem os diferentes requisitos das partes interessadas. Além disso, um avatar é caracterizado pelas suas capacidades de processar uma identidade única; comunicar de forma eficaz com seu ambiente; criar, acessar, transferir e operar informações sobre si mesmo; implantar um idioma para exibir suas características e requisitos; e de participar ou fazer decisões relevantes para o seu próprio objetivo (Wong et al.,

2002; Hribernik et al., 2006).

### **2.2.1.3 Interrelação entre Gêmeos Digitais e Avatares**

Ambos representam entidades no mundo virtual, sendo o Gêmeo Digital mais focado na replicação detalhada de produtos físicos e sistemas, enquanto o Avatar se concentra na representação gráfica de partes interessadas e informações relacionadas ao produto. Ambos desempenham papéis significativos em fornecer uma visão virtual e interativa ao longo do ciclo de vida, seja para simulação, análise de dados ou interações no ambiente virtual. Os Gêmeos Digitais podem ser vistos como uma categoria mais abrangente que inclui representações virtuais de produtos e sistemas, enquanto os Avatares são específicos para as partes interessadas e suas interações no ambiente virtual.

### **2.2.2 Mimetismo Virtual**

O mimetismo de uma pessoa do ambiente real para o ambiente virtual, na criação de um gêmeo digital, é um processo complexo que envolve a replicação detalhada e precisa das características físicas e comportamentais de um indivíduo. Esse procedimento busca não apenas reproduzir a aparência externa, mas também capturar a dinâmica e nuances do comportamento da pessoa.

O primeiro passo é a coleta de dados por meio de tecnologias avançadas, como escaneamento 3D, que permitem a criação de uma representação tridimensional fiel do corpo e das características físicas da pessoa. Sensores, câmeras e outros dispositivos capturam não apenas a forma estática, mas também os movimentos.

A utilização de tecnologias como o body tracking contribui para a captura dinâmica dos movimentos, permitindo que o gêmeo digital reproduza com precisão as ações e gestos da pessoa no ambiente real. Esse acompanhamento mimético é fundamental para proporcionar uma experiência autêntica e imersiva no ambiente virtual.

Além disso, o processo de escaneamento e mimetismo envolve a consideração de detalhes, como texturas da pele, expressões faciais em diferentes contextos e até mesmo a reação a estímulos específicos.

O gêmeo digital, assim criado, pode ser aplicado em uma variedade de contextos, desde ambientes de entretenimento e jogos virtuais até simulações para treinamento profissional. Ele representa uma ponte entre o mundo real e o virtual, permitindo a interação e a representação virtual de uma pessoa de forma fiel e dinâmica. Esse processo avançado de mimetismo contribui para a criação de experiências mais envolventes e personalizadas no ambiente digital.

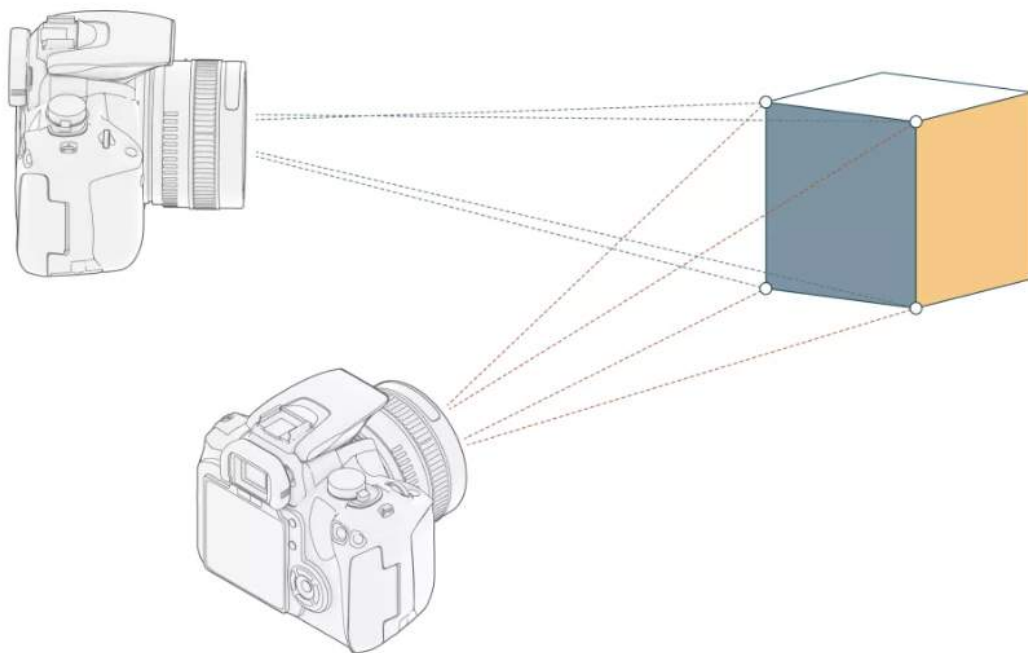
### 2.2.3 Triangulação

A triangulação no caso do escaneamento representa um alinhamento de câmeras e fontes de luz para ter a visão em 3 dimensões do objeto escaneado. Sendo que a triangulação pode ser feita com laser, ou como no caso do presente trabalho, com câmeras. O objetivo é obter as coordenadas tridimensionais do objeto.

Existem diversos métodos de triangulação. O que foi utilizado no projeto em questão, é a "Triangulação na fotogrametria".

Fotogrametria é o processo de tirar medidas confiáveis de fotografias. Com imagens sobrepostas suficientes fornecendo as informações espaciais necessárias, é possível reconstruir um modelo 3D de um objeto ou cena inteira. Esse método infere as dimensões de uma cena usando várias imagens sobrepostas tiradas de diferentes posições e ângulos conforme Fig. 2.1.

Figura 2.1 – Triangulação - fotogrametria



Fonte: Mubanga (2022)

Em uma câmera, todos os pontos em 3 dimensões, exceto o centro de projeção, criam uma linha única que passa pelo centro de projeção, e que caso não seja paralela ao plano da imagem, deve interceptar o plano em um único ponto. Para resumir a luz percorre o caminho indo de um projetor até uma câmera, ao longo de uma linha que conecta o ponto em três dimensões com a perspectiva em 2D projetada no centro do plano da imagem (TAUBIN; MORENO; LANMAN, 2014). Um exemplo de triangulação pode ser visto no trabalho de Franca et al. (2005) no qual se faz uso de um encoder, uma câmera e um laser para se ter três pontos do objeto escaneado.

Ter imagens sobrepostas é a chave para a fotogrametria. Ao identificar os mesmos pontos em várias imagens e levar em consideração parâmetros como a posição e orientação da câmera para cada fotografia, sua distância focal, distorção da lente e outras variáveis, é possível determinar onde esses pontos estavam localizados no espaço 3D. Isso se chama triangulação (MUBANGA, 2022).

A triangulação funciona identificando pontos comuns em imagens sobrepostas e determinando suas posições no espaço 3D em relação às posições conhecidas da câmera. Quando um ponto é identificado em pelo menos duas fotos tiradas de diferentes locais conhecidos, podemos traçar linhas imaginárias das duas posições da câmera na direção daquele ponto. Em seguida, determinamos matematicamente onde as linhas se cruzam. As linhas convergentes nos fornecem as coordenadas XYZ do ponto alvo. E com pontos suficientes, podemos construir um modelo da cena (MUBANGA, 2022).

#### 2.2.4 Nuvem de pontos

O modelo geométrico da nuvem de pontos é a representação mais básica obtida diretamente pelo escâner 3D. Cada ponto que forma a nuvem de pontos é representado por suas coordenadas cartesianas  $(x, y, z)$  e um ou mais atributos associados ao mesmo (GROETELAARS; AMORÍN, 2011).

O registro dos dados consiste em posicionar várias cenas, ou imagens coletadas pelo sensor, em único arquivo com o mesmo referencial. E então, com a associação das  $n$  imagens, com área de sobreposição acima de 30%, realizar operações para otimizar as nuvens. As operações consistem em fazer filtragem dos pontos relevantes e re-amostragem, eliminando informações redundantes.

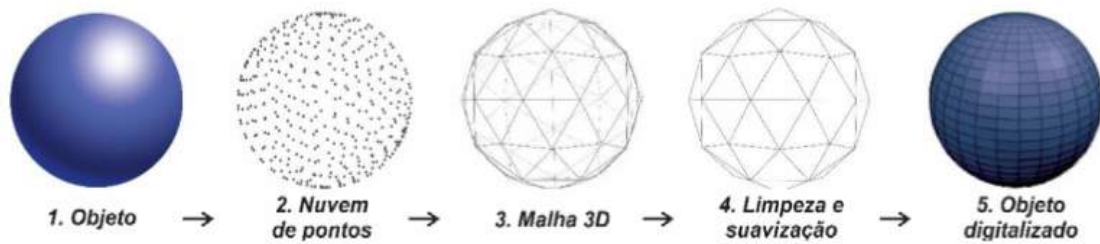
#### 2.2.5 Malha

Ao unirmos os pontos de uma nuvem de maneira coesa e coerente, obtemos um objeto que pode ser manipulado de acordo com o que se pretende analisar. No geral, definimos uma malha como uma matriz de conexões que compõe um grafo. Neste caso, as coordenadas desses nós são obtidas através do escaneamento, e as conexões através de algoritmos de reconstrução que estimam as conexões entre os nós.

O processo de obtenção de um objeto tridimensional consiste em duas etapas: no processamento da nuvem de pontos em uma rede de conexões, i.e., uma malha tridimensional, e no pós-processamento dessa malha bruta onde serão feitas suavizações, preenchimento de buracos, e eventuais decimações ou sobreamostragens. O processo de confecção da malha 3D pode ser visto na Fig. 2.2.

A qualidade final do escaneamento é influenciada por diversos fatores, indo desde a qualidade do sensor usado, da precisão dele e tecnologia usada, além de fatores de hardware,

Figura 2.2 – Processo de obtenção de malha 3D



Fonte: Sierra e Okimoto (2021)

cenário usado e protocolo de coleta (SIERRA; OKIMOTO, 2021). Por esses e outros fatores, a malha escaneada sempre irá apresentar algum tipo de erro, sendo pequeno ou não. Por isso existem diversas técnicas para esse processo de limpeza.

### 2.2.6 NuiTrack

O NuiTrack utilizado é um middleware de rastreamento do corpo 3D(3D BODY (SKELETAL) TRACKING MIDDLEWARE). Inspirado inicialmente no Microsoft Kinect, a missão do NuiTrack é fornecer uma base sólida de rastreamento esquelético para a próxima geração de aplicativos analíticos e imersivos além de qualquer plataforma ou hardware específico. Como um “Kinect para qualquer coisa”.

Um framework voltado para o desenvolvimento de aplicativos e soluções que envolvem rastreamento de corpo humano (body tracking) e interação gestual em tempo real. Projetado para ser utilizado em diversas plataformas, o NuiTrack oferece uma série de recursos que o tornam uma escolha atrativa em aplicações relacionadas à visão computacional e realidade virtual. Aqui estão alguns aspectos essenciais do framework:

- **Rastreamento de Corpo e Gestos:** se destaca por sua capacidade de rastreamento preciso do corpo humano em tempo real, permitindo a detecção e acompanhamento de movimentos articulares. Além disso, oferece suporte para o reconhecimento de gestos, possibilitando interações intuitivas e naturais.
- **Multiplataforma:** desenvolvido para ser multiplataforma, o NuiTrack é compatível com uma variedade de sistemas operacionais, incluindo Windows, Linux e Android. Isso proporciona flexibilidade aos desenvolvedores para criar aplicativos que atendam a diferentes ambientes e dispositivos.
- **Integração com Unity e Outras Plataformas:** projetado para integração fácil com várias plataformas de desenvolvimento, sendo especialmente compatível com o Unity, um po-

pular motor de jogo e ambiente de desenvolvimento. Essa integração facilita a criação de experiências interativas em ambientes virtuais.

- **Suporte para Diferentes Sensores:** oferece suporte para uma variedade de sensores 3D, incluindo câmeras RGB, câmeras de profundidade e sensores LiDAR. Essa flexibilidade permite que os desenvolvedores escolham os dispositivos mais adequados às suas necessidades específicas.
- **Aplicações em Diversos Setores:** o framework encontra aplicação em vários setores, desde jogos e entretenimento até educação, saúde e treinamento profissional. Sua capacidade de rastreamento preciso e reconhecimento de gestos o torna valioso em uma ampla gama de cenários.
- **Desenvolvimento Ativo e Suporte da Comunidade:** é mantido ativamente, com atualizações regulares e suporte contínuo. Além disso, a existência de uma comunidade ativa de desenvolvedores contribui para a resolução de problemas, compartilhamento de conhecimento e crescimento do ecossistema do framework.

Em resumo, o NuiTrack oferece um conjunto abrangente de recursos para o rastreamento de corpo e gestos, proporcionando aos desenvolvedores as ferramentas necessárias para criar aplicações interativas e envolventes em diversos setores e plataformas.

### 2.2.7 Distância de Chamfer

A distância de Chamfer é uma métrica usada para avaliar a semelhança entre dois conjuntos de pontos. Dados dois conjuntos de pontos A e B, a distância de Chamfer é definida como a soma das distâncias de cada ponto em A ao seu vizinho mais próximo em B, mais a soma das distâncias de cada ponto em B ao seu vizinho mais próximo em A. Ela é utilizada em uma variedade de aplicações, incluindo visão computacional, robótica e gráficos computacionais. A distância de Chamfer é particularmente útil no registro de nuvens de pontos, onde o objetivo é alinhar duas ou mais nuvens de pontos. Ao minimizar a distância de Chamfer entre os conjuntos de pontos, pode-se encontrar o melhor alinhamento entre eles.

No contexto de aprendizado profundo, a distância de Chamfer é frequentemente usada como uma função de perda em tarefas de geração de nuvens de pontos. O objetivo nessas tarefas é gerar uma nuvem de pontos que seja semelhante a uma nuvem de pontos alvo. A perda de distância de Chamfer incentiva a nuvem de pontos gerada a ser o mais próxima possível da nuvem de pontos alvo.

Figura 2.3 – Distância de Chamfer

$$\text{chamfer}(P_1, P_2) = \frac{1}{2n} \sum_{i=1}^n |x_i - \text{NN}(x_i, P_2)| + \frac{1}{2m} \sum_{j=1}^m |x_j - \text{NN}(x_j, P_1)|$$

Fonte: [www.fwilliams.info/point-cloud-utils/sections/shape\\_metrics/](http://www.fwilliams.info/point-cloud-utils/sections/shape_metrics/)

De acordo com a fórmula da Figura 2.3, a distância do Chamfer entre duas nuvens de pontos P1 e P2 é definida como a distância média entre pares de vizinhos mais próximos entre P1 e P2 e NN é a função vizinha mais próxima.

## 2.3 Trabalhos Relacionados

O uso de body tracking e sensores RGB-D tem sido uma área de interesse em pesquisa por muitos anos. A literatura existente demonstra uma variedade de aplicações e objetivos distintos em projetos relacionados. Embora os propósitos e objetivos possam diferir, esses estudos fornecem uma base sólida de conhecimento e metodologias relevantes para o atual projeto de pesquisa.

Um estudo comparativo recente investigou o desempenho de diferentes câmeras e algoritmos de rastreamento de esqueleto, destacando as vantagens das câmeras ZED-2i sobre a RealSense. Os resultados sugerem a viabilidade dessas tecnologias para aprimorar a precisão da captura de movimento N. Meshurer A. (2023).

Em outro estudo, enfatiza-se a importância dos avatares pessoais na Realidade Virtual (RV), propondo um método automático para ajustar as dimensões dos avatares. Essa abordagem,



baseada em exercícios do usuário, mostra-se eficaz na redução de discrepâncias em comparação com métodos manuais [L. Ceballos V. \(2023\)](#).

Utilizando Kinect um trabalho de 2022 sobre rastreamento de esqueleto, utilizando a fusão de dados de múltiplos Kinects, destaca a importância da precisão e robustez no contexto de simulações de montagem assistidas por Realidade Virtual (VR). O algoritmo proposto neste trabalho foca na avaliação da confiança no rastreamento, considerando erros relacionados à posição e auto-ocultação do corpo humano. Os resultados indicam que o algoritmo desenvolvido supera outras abordagens similares, demonstrando uma melhor performance na fusão de dados dos múltiplos Kinects [Y. Chang F. \(2022\)](#).

Além de trabalhos voltados para avatares digitais e o entretenimento, também com o uso do conceito do body tracking que destaca a importância dos sistemas de detecção de quedas, especialmente para idosos, visando prevenir lesões graves ou mesmo fatais. O trabalho avalia o desenvolvimento de um sistema de detecção de quedas baseado no sensor Kinect da Microsoft, conhecido por sua capacidade de rastrear posições articulares. O estudo investiga o potencial do Kinect como uma ferramenta sofisticada para detecção de quedas [S. \(2014\)](#).

Mesmo alguns trabalhos de mais de dez anos atrás como o de 2011 que explorou as aplicações da tecnologia de câmera de detecção de profundidade, como o Kinect da Microsoft, em três áreas específicas: ambientes virtuais, jogos controlados por gestos e reabilitação de movimentos específicos. O trabalho desenvolveu middleware de código aberto para permitir que desenvolvedores criem aplicativos para essas áreas. Utilizando os sensores PrimeSense ou Kinect e o middleware desenvolvido, o estudo conseguiu rastrear completamente o corpo do usuário em tempo real, sem a necessidade de dispositivos de rastreamento adicionais. Isso abriu caminho para uma série de aplicações interativas e de treinamento que podem ser utilizadas em diversas áreas, incluindo a reabilitação física [B. Rizzo S. \(2011\)](#).

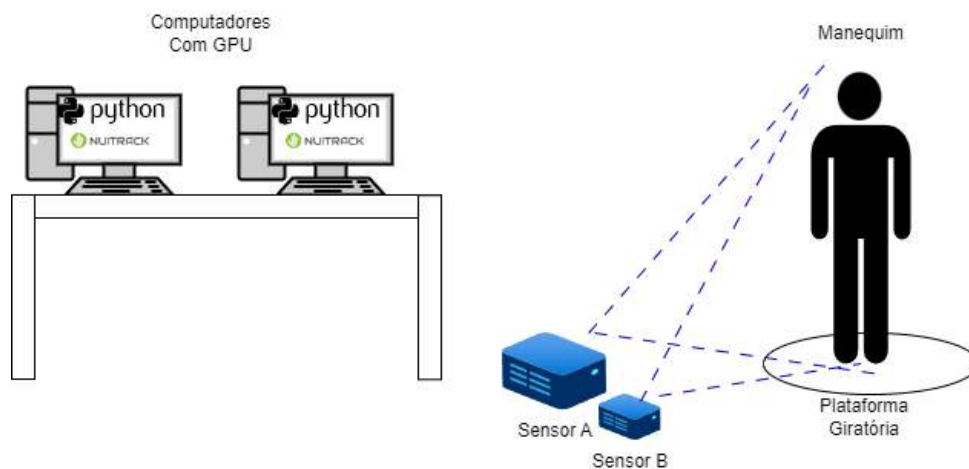
Por fim, utilizando Vive Trackers, um estudo desenvolveu uma abordagem precisa e de baixa latência para o rastreamento corporal em RV. Essa tecnologia permite criar uma experiência imersiva, animando os movimentos do avatar de forma suave e precisa [P. Garcia-Agundez A. \(2019\)](#).

## 3 Metodologia

O presente projeto propõe comparar a eficácia do body tracking utilizando dois sensores distintos: o Kinect e o sensor Astral S em combinação com o software NuiTrack. Através de uma metodologia rigorosa, avaliaremos a qualidade da esqueletização de movimentos, a precisão, e a latência de resposta de cada sistema. A análise de distância de Chamfer será aplicada para determinar a concordância entre os dados capturados, permitindo assim identificar qual dos dois sensores oferece a melhor solução para a para estimação de poses antropométricas de avatares digitais. Esta abordagem detalhada visa otimizar a aplicação de tecnologias de captura de movimento para gêmeos digitais, equilibrando inovação tecnológica com acessibilidade financeira.

Este trabalho está dividido em cinco partes principais, estruturadas para abordar de forma abrangente a comparação entre duas tecnologias de captura de movimento, o Kinect e o NuiTrack em combinação com o sensor Astral S, no contexto da esqueletização de movimentos corporais:

Figura 3.1 – Diagrama de Configuração dos Componentes



Fonte: Autoria Própria

### 3.1 Componentes

#### 3.1.1 Sensores RGB-D

Os dois sensores utilizados foram o Orbbec ASTRA STEREO S U3 e o Microsoft Kinect. Um sensor com preço elevado e um com preço mais acessível.

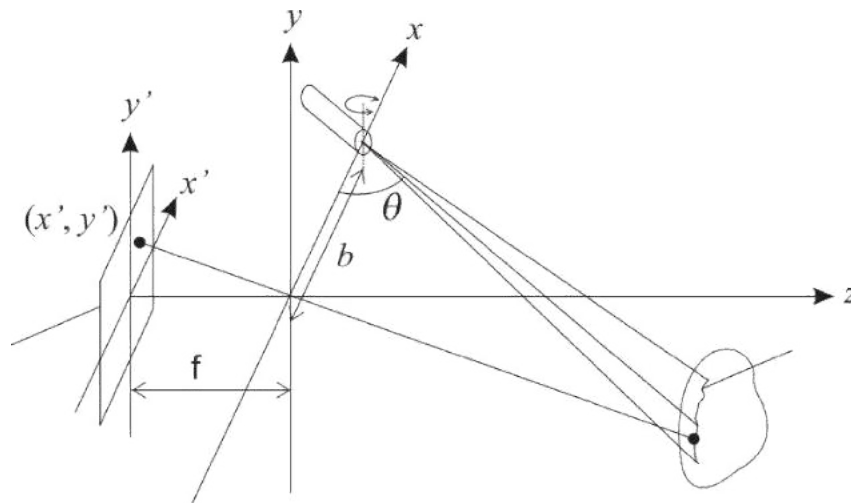
Ambos são sensores que capturam informações RGB (Red, Green, Blue) juntamente com dados de profundidade (Depth), proporcionando uma representação tridimensional do ambiente.

São essenciais em aplicações como realidade aumentada, interação humano-computador, robótica, e segurança.

### 3.1.1.1 Estimação de distância em sensores RGB-D

O processador de imagem interno de ambos os sensores Astra e Kinect utiliza a posição relativa dos pontos presentes no padrão de emissão para calcular a distância entre o plano câmera-laser e o plano do objeto, em uma técnica de triangulação, ilustrada na Figura 3.2.

Figura 3.2 – Geometria da técnica de triangulação: as lentes são localizadas na origem, com distância focal  $f$  para o plano da imagem e a linha base  $b$  entre o projetor e a câmera.



Fonte: Adaptado de (FRANCA et al., 2005)

Na técnica de triangulação, a projeção do laser forma um ângulo  $\theta$  com a linha base  $b$  e o ponto 3D  $(x, y, z)$  no espaço real é projetado para o pixel de uma imagem  $(x, y)$ , seguindo as equações abaixo:

$$x' = \frac{x \cdot f}{z} \quad (3.1)$$

$$y' = \frac{y \cdot f}{z} \quad (3.2)$$

Já as coordenadas reais de um ponto seguem as Equações a seguir:

$$x = \frac{b \cdot x'}{f \cdot \cot\theta - x'} \quad (3.3)$$

$$y = \frac{b \cdot y'}{f \cdot \cot\theta - x'} \quad (3.4)$$

$$z = \frac{b.z'}{f.\cot\theta - x'} \quad (3.5)$$

O valor  $\theta$  é dado pela posição do sensor, a distância focal  $f$  é calculada pela calibração da câmera e a linha base  $b$ , é a distância entre a câmera e o projetor de luz estruturada infra-vermelha do sensor. Em outros tipos de sensores a linha base é calculada posicionando a origem do laser em um ponto onde sua projeção coincide com o centro da imagem (apontando para um objeto de distância  $z$  conhecida), seguindo a equação abaixo.

$$b = \frac{z}{\tan\theta} \quad (3.6)$$

### 3.1.1.2 Sensor Orbbec ASTRA STEREO S U3

O Astra Stereo S U3 é a primeira e a mais compacta câmera 3D baseada na tecnologia Active Stereo IR da Orbbec e está equipada com o ASIC personalizado da Orbbec para processamento de profundidade de alta qualidade e conexão + alimentação por um único cabo USB 3.0. O Astra Stereo S U3 é utilizado para aplicações de curto alcance e alta precisão em varejo, seleção por robôs, medição de objetivos, identificação facial, etc. O SDK Orbbec OpenNI facilita a configuração e entrega dados altamente precisos e confiáveis em diversas condições de iluminação.

Projetado para ser um dispositivo versátil para captura de movimento e reconhecimento de gestos, é amplamente utilizado em pesquisa e desenvolvimento. Emprega tanto a luz estruturada quanto o time-of-flight (ToF), dependendo do modelo específico, para medir a profundidade. O ToF emite um pulso de luz e mede o tempo que ele leva para retornar.

Utilizado em áreas como robótica, análise de movimento, e interação imersiva em realidade virtual.

Figura 3.3 – Sensor ASTRA STEREO S U3



Fonte: Autoria Própria

### 3.1.1.3 Kinect

O Kinect é um sensor de movimentos desenvolvido para o Xbox 360 e Xbox One, junto com a empresa Prime Sense. A tecnologia incorpora câmeras RGB, projetores infravermelhos e detectores que mapeiam a profundidade através de cálculos estruturados de luz ou por tempo de voo. O sensor também possui um conjunto de microfones que se unem com o software e a inteligência artificial da Microsoft para permitir que o dispositivo realize reconhecimento de gestos em tempo real, reconhecimento de voz e detecção esquelética corporal de até quatro pessoas.

Utiliza uma combinação de câmera RGB, sensor de profundidade e microfones para permitir uma interação intuitiva através de gestos e comandos de voz. Além disso possui tecnologia de luz estruturada, que projeta um padrão de pontos infravermelhos e utiliza uma câmera IR para captar as distorções nesse padrão, determinando assim a distância dos objetos.

Utilizado em áreas como jogos e entretenimento até mesmo aplicações terapêuticas e de monitoramento de movimento.

Figura 3.4 – Microsoft Kinect



Fonte: Autoria Própria

### 3.1.2 Placa de vídeo

Para a realização do projeto foram utilizados dois desktop com placa de vídeo RTX para processamento gráfico da GPU, tendo em vista renderização 3D de maneira mais eficiente e com menor latência. Além disso o software NuiTrack tem GPU como requisito de hardware para executar o código e realizar o body tracking.

A placa de vídeo usada para o Setup A foi a NVIDIA Titan XP(12GB) e para o Setup B a NVIDIA RTX 3060(12GB).

Figura 3.5 – Placa de vídeo Titan XP



Fonte: Autoria Própria

Figura 3.6 – Placa de vídeo Rtx 3060



Fonte: Autoria Própria

### 3.1.3 Plataforma giratória

A plataforma giratória utilizada tem capacidade de 150Kg e velocidade de rotação de 2,5rpm. Na Fig. 3.7, podemos ver a plataforma giratória utilizada em laboratório.

#### 3.1.3.1 Controle de Velocidade

Ao nos debruçarmos sobre objetivos do escaneamento 3D, recaímos em dois problemas: a quantidade de dados e o tempo de escaneamento. Ao rotacionarmos o objeto lentamente, podemos obter uma nuvem de pontos mais detalhada, uma vez que as pequenas variações poderão ser detectadas, contudo, o tempo de escaneamento será elevado. Por outro lado, ao rotacionarmos rapidamente o objeto, perderemos nuances de sua superfície obtendo um escaneamento mais simples, embora mais rápido.

Figura 3.7 – Plataforma giratória utilizada na rotação do manequim



Fonte: Autoria Própria

Para obtermos uma taxa de variação constante e ajustável precisamos implementar um controle da velocidade do motor. Dado que os corpos submetidos ao escaneamento possuem momentos de inércia diferentes, possíveis trepidações e eventuais oscilações do fornecimento de energia, faz-se necessário um sistema de controle por retroalimentação.

Por tratar-se de um sistema relativamente simples, recorreremos ao uso do controle Proporcional, Integral e Derivativo (PID) afim de controlarmos o sistema motor-plataforma. Assim, desenvolvemos um sistema de controle PID digital. Neste sistema de controle, variamos a velocidade do motor em função da tensão de alimentação que pode ser ajustada pelo PWM (do inglês, *pulse with modulation*) aplicado à ponte H.

A velocidade angular da plataforma pode ser obtida com a relação de transmissão entre a plataforma e o motor. Dessa forma, a cada atualização da posição angular do motor era somado a um acumulador a posição total da plataforma e com isso, calculávamos a derivada primeira de maneira discreta. Mais uma vez utilizamos do processamento paralelo para atualizar o controlador e a atuação dele sobre o sistema.

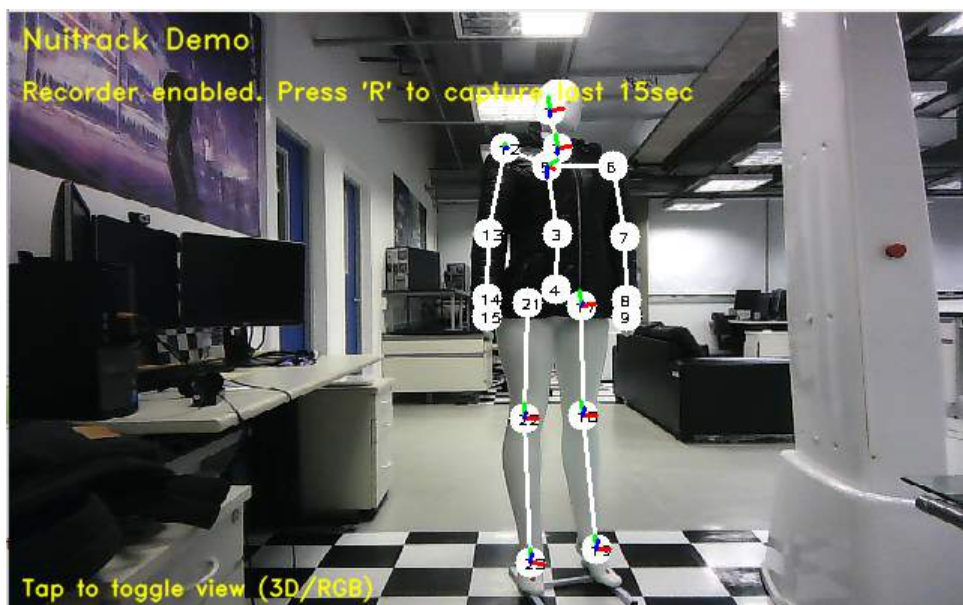
Para realizar o ajuste dos parâmetros de controle  $K_p$ ,  $K_i$  e  $K_d$  utilizamos um método heurístico implementado por bissecção. Começamos por ajustar o ganho proporcional e, enquanto monitorávamos o erro estacionário, aumentávamos o ganho integral. Por fim, incluímos um ganho derivativo para lidar com oscilações repentinas de velocidade da plataforma, e melhorar o tempo de resposta ao se alterar a velocidade via comunicação serial.

Embora as constantes finais tenham atendido aos nossos requisitos, ainda observava-se uma variação errática da velocidade em função de ruídos de leitura do transdutor de posição angular. Para reduzir os efeitos dos ruídos de alta frequência, implementamos um filtro passa baixas.

### 3.1.4 NuiTrack

Conforme comentado anteriormente o NuiTrack é um framework voltado para o desenvolvimento de aplicativos e soluções que envolvem rastreamento de corpo humano (body tracking) e interação gestual em tempo real. Projetado para ser utilizado em diversas plataformas, o NuiTrack oferece uma série de recursos que o tornam uma escolha atrativa em aplicações relacionadas à visão computacional e realidade virtual. A versão completa do software é paga, mas para realizar o experimento precisamos utilizar a versão de licença universitária e o SDK do framework em python para melhor obter os dados e realizar a comparação.

Figura 3.8 – Software NuiTrack capturando imagens do manequim.



Fonte: Autoria Própria

### 3.1.5 Manequim

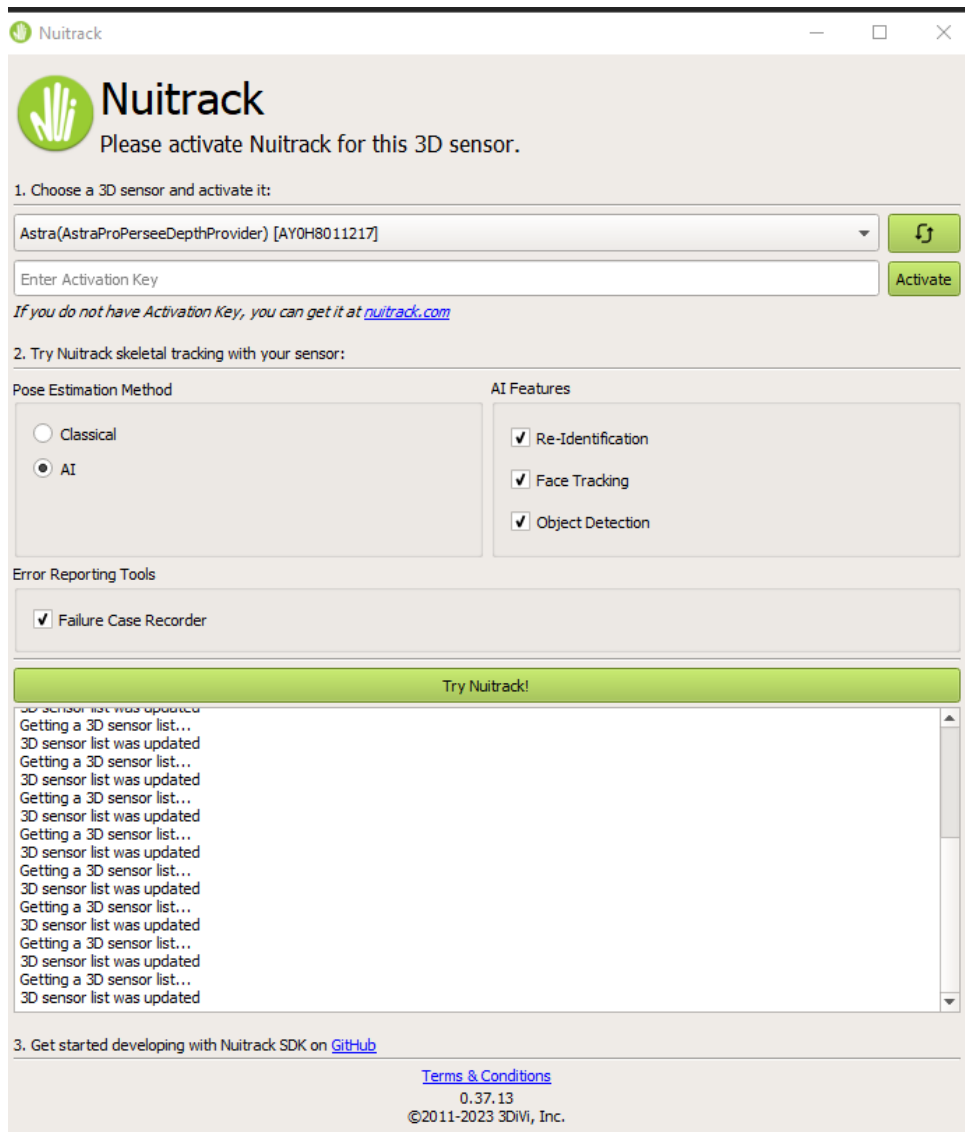
Utilizamos um manequim em detrimento de um participante humano, visando uma maior consistência e controle sobre as variáveis do experimento. Permitindo a captura precisa das coordenadas X, Y, e Z das juntas do manequim, fornecendo um conjunto de dados uniforme e repetível, essencial para uma análise comparativa rigorosa entre os sensores testados.

## 3.2 Preparação e Configuração do Experimento

No desenvolvimento deste experimento em ambos os sensores, um manequim foi empregado como elemento central para a coleta de dados. Optamos pelo uso de um manequim em detrimento de um participante humano, visando uma maior consistência e controle sobre as variáveis do experimento. Esse método permitiu a captura precisa das coordenadas X, Y, e Z das



Figura 3.9 – Software NuiTrack



Fonte: Autoria Própria

juntas do manequim, fornecendo um conjunto de dados uniforme e repetível, essencial para uma análise comparativa rigorosa entre os sensores testados.

A decisão de usar um manequim surgiu da necessidade de eliminar as variações inerentes aos movimentos humanos, que podem introduzir inconsistências significativas devido à dificuldade em reproduzir exatamente os mesmos movimentos em múltiplas tentativas. Com o manequim, foi possível estabelecer um padrão de movimento preciso e replicável, garantindo que as diferenças observadas na captura de dados refletissem unicamente as características e capacidades dos sensores em análise, não sendo influenciadas por variações na execução dos movimentos.

Além disso, o uso do manequim facilitou a execução de movimentos específicos nas coordenadas espaciais X, Y, e Z, permitindo uma avaliação detalhada da capacidade de cada sensor

Figura 3.10 – Manequim em plataforma giratória.



Fonte: Autoria Própria

Figura 3.11 – Tabela de Setup A e B

	SETUP A	SETUP B
Sensor	Astra Stereo S U3	Microsoft Kinect 1.0
GPU	NVIDIA Titan XP (12GB)	NVIDIA RTX 3060 (12 GB)

Fonte: Autoria Própria

em detectar e registrar com precisão os pontos de articulação. Essa abordagem metodológica não apenas reforçou a objetividade e a confiabilidade dos resultados obtidos, como também contribuiu para uma análise comparativa mais fundamentada entre o sensor Orbbec e seu(s) concorrente(s), estabelecendo uma base sólida para conclusões sobre a eficácia dos sensores em aplicações de captura de movimento controlado.

### 3.3 Captura de Poses Antropométrica

No experimento, foi realizada a captura de poses antropométricas utilizando dois sensores e seus respectivos setups, ambos integrados com o software NuiTrack, que permite a criação de uma nuvem de pontos semelhante para ambos os dispositivos. A configuração foi meticulosamente planejada para garantir que os sensores estivessem posicionados de maneira a maximizar a semelhança na captura dos dados de profundidade, fundamental para um estudo comparativo preciso.

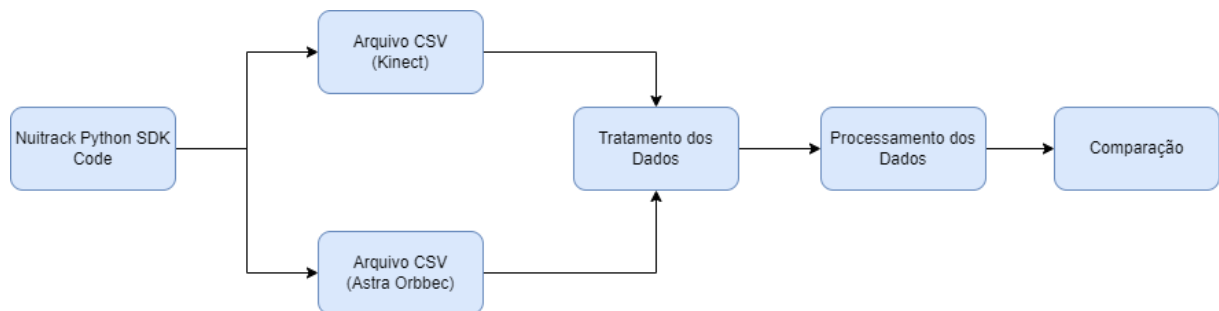
A SDK do NuiTrack foi utilizada para programar um código em Python de autoria própria

Figura 3.12 – Configuração Sensores para Captura



Fonte: Autoria Própria

Figura 3.13 – Diagrama de captura de dados.



Fonte: Autoria Própria

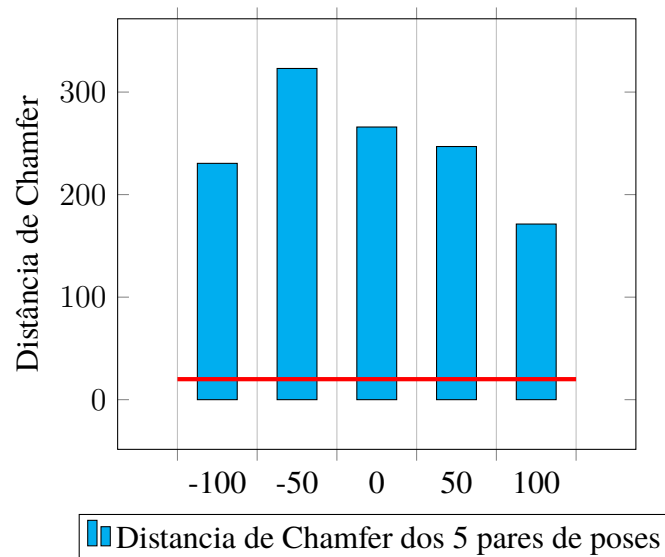
que foi imprescindível para obter as coordenadas de cada junta do esqueleto das poses capturadas. Os dados foram exportados em arquivo CSV para melhor análise e processamento, proporcionando um terreno comum para avaliar a eficácia de cada sensor em capturar movimentos complexos do corpo humano em diversas poses antropométricas.

## 4 Resultados

Pela Figura 4.1 descartamos a hipótese de que ambos os sensores teriam capacidades similares para estimação de poses através do cálculo da distância de Chamfer, realizado em cinco conjuntos de pares de poses. Cada conjunto foi escolhido manualmente em função da proximidade inicial das coordenadas X da cabeça entre cada par.

A linha vermelha na faixa de 20 unidades adimensionais indica o limiar máximo de erro aceitável (cerca de 1%) para torerância de eventual discrepância dos sensores na determinação da pose a partir da malha 3d. A Table 4.1 que segue apresenta quais linhas de dados da planilha de aquisição foram escolhidas para a comparação dos pares.

Figura 4.1 – Resultado final da comparação entre os sensores



Fonte: Autoria Própria

	Astra	Kinect
-100	1383	2128
-50	766	1221
0	2096	1665
50	6	37
100	1860	1800

Tabela 4.1 – Indicativo das linhas da coleta de dados utilizadas na comparação.

## 5 Conclusão

Dada as amplitudes dos resultados obtidos na seção anterior, derrubamos a hipótese de similaridade entre os sensores.

Quanto mais próximo de zero tivessem sido os valores calculados nas distâncias de Chamfer, mais próximo seriam os resultados de ambos os sensores e mais similares eles seriam entre si para esse tipo de aplicação, de determinação de pose.

Em sendo o fato de que os valores médios medidos tinham sua precisão em escala unitária ou em dezenas, um valor de desvio da ordem mais de uma centena certamente corrobora o fato de que ambos os sensores não podem ser considerados similares para uso nessa função específica de estimar a postura com finalidade de aplicação em controle de poses em gêmeos digitais.

# Referências

- B. RIZZO S., C. C.-y. S. E. A. . B. M. L. Markerless full body tracking: Depth-sensing technology within virtual environments. 2011. Disponível em: <<http://deadnet.se:8080/ict.usc.edu/pubs/Markerless%20Full%20Body%20Tracking-%20Depth-Sensing%20Technology%20within%20Virtual%20Environments.pdf>>.
- COSTELLO, G. O. K. Gartner survey reveals digital twins are entering mainstream use. Gartner, 2019. Disponível em: <<https://www.gartner.com/en/newsroom/press-releases/2019-02-20-gartner-survey-reveals-digital-twins-are-entering-mai>>.
- D'MELLO, A. Global digital twin market size to exceed \$20bn by 2025 | iot now news reports. IoT Now News – How to run an IoT enabled business, 2019. Disponível em: <<https://www.iot-now.com/2019/07/08/97220-global-digital-twin-market-size-exceed-20bn-2025/>>.
- FRANCA, J.; GAZZIRO, M.; IDE, A.; SAITO, J. A 3d scanning system based on laser triangulation and variable field of view. In: *IEEE International Conference on Image Processing 2005*. IEEE, 2005. Disponível em: <<https://doi.org/10.1109/icip.2005.1529778>>.
- GARCIA BIANCA GABALDE, G. B.-M. B. E. S. F. C. A. Aplicação de gêmeos digitais na indústria da construção: estado da arte. Universidade do Minho, p. 12, 2022. Disponível em: <<https://doi.org/10.21814/uminho.ed.32.44>>.
- GROETELAARS, N. J.; AMORÍN, A. Tecnologia 3d laser scanning: Características processos e ferramentas para manipulação de nuvens de pontos [3d laser scanning technology: characteristics, processes and point cloud tools]. CUMINCAD, 2011.
- JIANG, Y.; YIN, S.; LI, K.; LUO, H.; KAYNAK, O. Industrial applications of digital twins. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, The Royal Society, v. 379, n. 2207, p. 20200360, ago. 2021. Disponível em: <<https://doi.org/10.1098/rsta.2020.0360>>.
- L. CEBALLOS V., A. L. R. A. M. E. . P. N. P. J. Fitted avatars: automatic skeleton adjustment for self-avatars in virtual reality. *Virtual Reality*, 2023. Disponível em: <<https://doi.org/10.1007/s10055-023-00821-z>>.
- MUBANGA, K. What is photogrammetry. Artec 3D, 2022. Disponível em: <<https://www.artec3d.com/learning-center/what-is-photogrammetry>>.
- N. MESHURER A., K. M. P. Y. M. I. . E. Y. A. Comparative analysis of cameras and software tools for skeleton tracking. *bioRxiv*, 2023. Disponível em: <<https://doi.org/10.1101/2023.08.10.552434>>.
- P. GARCIA-AGUNDEZ A., K. R. G. S. . S. R. C. Real-time body tracking in virtual reality using a vive tracker. *International Journal of Distributed Sensor Networks*, 2019. Disponível em: <<https://doi.org/10.1007/s10055-018-0374-z>>.
- S., . D. S. S. Depth sensor based skeletal tracking evaluation for fall detection systems. 2014. Disponível em: <<https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=08670ecfa146027548b8cac3a148d7d6ebbe994e>>.

SIERRA, I. de S.; OKIMOTO, M. L. Proposta de protocolo de análise da malha escaneada (map). *Design e Tecnologia*, v. 11, n. 22, p. 83–92, 2021.

TAUBIN, G.; MORENO, D.; LANMAN, D. 3d scanning for personal 3d printing. In: *ACM SIGGRAPH 2014 Studio on - SIGGRAPH '14*. ACM Press, 2014. Disponível em: <<https://doi.org/10.1145/2619195.2656314>>.

Y. CHANG F., W. Y. H. Z. L. L. L. P. L. P. . Y. S. W. Multi-kinects fusion for full-body tracking in virtual reality-aided assembly simulation. *International Journal of Distributed Sensor Networks*, 2022. Disponível em: <<https://doi.org/10.1177/15501329221097591>>.

# **A Apêndices - Código fonte**

## **A.0.1 Nuitrack SDK Script**



```

1  from PyNuitrack import py_nuitrack
2  import cv2
3  from itertools import cycle
4  import numpy as np
5  import time
6
7  def update_global_timestamp(start_time):
8      global global_timestamp
9      # Atualiza o global_timestamp com o tempo decorrido em microssegundos
10     global_timestamp = int((time.time() - start_time) * 1e6)
11
12     def draw_face(image):
13         if not data_instance:
14             return
15         for instance in data_instance["Instances"]:
16             line_color = (59, 164, 225)
17             text_color = (59, 255, 255)
18             if 'face' in instance.keys():
19                 bbox = instance["face"]["rectangle"]
20             else:
21                 return
22             x1 = (round(bbox["left"]), round(bbox["top"]))
23             x2 = (round(bbox["left"]) + round(bbox["width"]), round(bbox["top"]))
24             x3 = (round(bbox["left"]), round(bbox["top"]) + round(bbox["height"]))
25             x4 = (round(bbox["left"]) + round(bbox["width"]), round(bbox["top"]) + round(
bbox["height"]))
26             cv2.line(image, x1, x2, line_color, 3)
27             cv2.line(image, x1, x3, line_color, 3)
28             cv2.line(image, x2, x4, line_color, 3)
29             cv2.line(image, x3, x4, line_color, 3)
30             cv2.putText(image, "User {}".format(instance["id"]),
31 x1, cv2.FONT_HERSHEY_SIMPLEX, 1, text_color, 2, cv2.LINE_AA)
32             cv2.putText(image, "{} {}".format(instance["face"]["gender"],int(instance[
"face"]["age"]["years"])),
33 x3, cv2.FONT_HERSHEY_SIMPLEX, 1, text_color, 2, cv2.LINE_AA)
34
35     def draw_skeleton(image):
36         point_color = (59, 164, 0)
37         for skel in data.skeletons:
38             for el in skel[1:]:
39                 x = (round(el.projection[0]), round(el.projection[1]))
40                 cv2.circle(image, x, 8, point_color, -1)
41
42     # Adiciona a variável global para o timestamp
43     global_timestamp = 0
44     nuitrack = py_nuitrack.Nuitrack()
45     nuitrack.init()
46
47     # Captura o timestamp de início para calcular o tempo decorrido
48     start_time = time.time()
49
50     devices = nuitrack.get_device_list()
51     for i, dev in enumerate(devices):
52         print(dev.get_name(), dev.get_serial_number())
53         if i == 0:
54             dev.activate("license:55157:XuPgpkguyKV9oPG6") #you can activate device using
python api
55             print(dev.get_activation())
56             nuitrack.set_device(dev)
57
58
59     print(nuitrack.get_version())
60     print(nuitrack.get_license())
61
62     nuitrack.create_modules()
63     nuitrack.run()
64
65     modes = cycle(["depth", "color"])
66     mode = next(modes)
67     print("Timestamp, "
68         "Cabeça X,Cabeça Y,Cabeça Z,"
69         "Pescoço X,Pescoço Y,Pescoço Z,"

```

```

70     "Tronco X,Tronco Y,Tronco Z,"
71     "Cintura X,Cintura Y,Cintura Z,"
72     "Ombro Esquerdo X,Ombro Esquerdo Y,Ombro Esquerdo Z,"
73     "Ombro Direito X,Ombro Direito Y,Ombro Direito Z,"
74     "Cotovelo Esquerdo X,Cotovelo Esquerdo Y,Cotovelo Esquerdo Z,"
75     "Cotovelo Direito X,Cotovelo Direito Y,Cotovelo Direito Z,"
76     "Pulso Esquerdo X,Pulso Esquerdo Y,Pulso Esquerdo Z,"
77     "Pulso Direito X,Pulso Direito Y,Pulso Direito Z,"
78     "Quadril Esquerdo X,Quadril Esquerdo Y,Quadril Esquerdo Z,"
79     "Quadril Direito X,Quadril Direito Y,Quadril Direito Z,"
80     "Joelho Esquerdo X,Joelho Esquerdo Y,Joelho Esquerdo Z,"
81     "Joelho Direito X,Joelho Direito Y,Joelho Direito Z,"
82     "Tornozelo Esquerdo X,Tornozelo Esquerdo Y,Tornozelo Esquerdo Z,"
83     "Tornozelo Direito X,Tornozelo Direito Y,Tornozelo Direito Z,"
84     "Mao Esquerda X,Mao Esquerda Y,Mao Esquerda Z,"
85     "Mao Direita X,Mao Direita Y,Mao Direita Z,"
86     "Colarinho Esquerdo X,Colarinho Esquerdo Y,Colarinho Esquerdo Z,"
87     "Colarinho Direito X,Colarinho Direito Y,Colarinho Direito Z,"
88     )
89 while 1:
90     key = cv2.waitKey(1)
91     nitrack.update()
92     data = nitrack.get_skeleton()
93     # print(f"Timestamp FORA: {global_timestamp}")
94     data_instance = nitrack.get_instance()
95
96     if data is not None and hasattr(data, 'skeletons'):
97         # Atualiza o timestamp global com o tempo decorrido
98         update_global_timestamp(start_time)
99         for skeleton in data.skeletons:
100             try:
101
102                 joint_types = [
103                     "head",
104                     "neck",
105                     "torso",
106                     "waist",
107                     "left_collar",
108                     "left_shoulder",
109                     "left_elbow",
110                     "left_wrist",
111                     "left_hand",
112                     "right_collar",
113                     "right_shoulder",
114                     "right_elbow",
115                     "right_wrist",
116                     "right_hand",
117                     "left_hip",
118                     "left_knee",
119                     "left_ankle",
120                     "right_hip",
121                     "right_knee",
122                     "right_ankle"
123                 ]
124
125
126                 # Substitua os prints dentro do try block pelo seguinte:
127                 head_x, head_y, head_z = skeleton.head.real
128                 neck_x, neck_y, neck_z = skeleton.neck.real
129                 torso_x, torso_y, torso_z = skeleton.torso.real
130                 waist_x, waist_y, waist_z = skeleton.waist.real
131                 left_shoulder_x, left_shoulder_y, left_shoulder_z = skeleton.
132                 left_shoulder.real
133                 right_shoulder_x, right_shoulder_y, right_shoulder_z = skeleton.
134                 right_shoulder.real
135                 left_elbow_x, left_elbow_y, left_elbow_z = skeleton.left_elbow.real
136                 right_elbow_x, right_elbow_y, right_elbow_z = skeleton.right_elbow.
137                 real
138                 left_wrist_x, left_wrist_y, left_wrist_z = skeleton.left_wrist.real
139                 right_wrist_x, right_wrist_y, right_wrist_z = skeleton.right_wrist.
140                 real
141                 left_hip_x, left_hip_y, left_hip_z = skeleton.left_hip.real

```

```

138         right_hip_x, right_hip_y, right_hip_z = skeleton.right_hip.real
139
140         left_knee_x, left_knee_y, left_knee_z = skeleton.left_knee.real
141         right_knee_x, right_knee_y, right_knee_z = skeleton.right_knee.real
142         left_ankle_x, left_ankle_y, left_ankle_z = skeleton.left_ankle.real
143         right_ankle_x, right_ankle_y, right_ankle_z = skeleton.right_ankle.
144         real
145         left_hand_x, left_hand_y, left_hand_z = skeleton.left_hand.real
146         right_hand_x, right_hand_y, right_hand_z = skeleton.right_hand.real
147         left_collar_x, left_collar_y, left_collar_z = skeleton.left_collar.
148         real
149         right_collar_x, right_collar_y, right_collar_z = skeleton.right_collar
150         .real
151     print(f"{global_timestamp},"
152           f"{head_x},{head_y},{head_z},"
153           f"{neck_x},{neck_y},{neck_z},"
154           f"{torso_x},{torso_y},{torso_z},"
155           f"{waist_x},{waist_y},{waist_z},"
156           f"{left_shoulder_x},{left_shoulder_y},{left_shoulder_z},"
157           f"{right_shoulder_x},{right_shoulder_y},{right_shoulder_z},"
158           f"{left_elbow_x},{left_elbow_y},{left_elbow_z},"
159           f"{right_elbow_x},{right_elbow_y},{right_elbow_z},"
160           f"{left_wrist_x},{left_wrist_y},{left_wrist_z},"
161           f"{right_wrist_x},{right_wrist_y},{right_wrist_z},"
162           f"{left_hip_x},{left_hip_y},{left_hip_z},"
163           f"{right_hip_x},{right_hip_y},{right_hip_z},"
164           f"{left_knee_x},{left_knee_y},{left_knee_z},"
165           f"{right_knee_x},{right_knee_y},{right_knee_z},"
166           f"{left_ankle_x},{left_ankle_y},{left_ankle_z},"
167           f"{right_ankle_x},{right_ankle_y},{right_ankle_z},"
168           f"{left_hand_x},{left_hand_y},{left_hand_z},"
169           f"{right_hand_x},{right_hand_y},{right_hand_z},"
170           f"{left_collar_x},{left_collar_y},{left_collar_z},"
171           f"{right_collar_x},{right_collar_y},{right_collar_z}"
172     )
173
174     except AttributeError as e:
175         print(f"Erro ao acessar dados do esqueleto: {e}")
176
177     img_depth = nitrack.get_depth_data()
178     if img_depth.size:
179         cv2.normalize(img_depth, img_depth, 0, 255, cv2.NORM_MINMAX)
180         img_depth = np.array(cv2.cvtColor(img_depth, cv2.COLOR_GRAY2RGB), dtype=np.
181         uint8)
182         img_color = nitrack.get_color_data()
183         draw_skeleton(img_depth)
184         draw_skeleton(img_color)
185         draw_face(img_depth)
186         draw_face(img_color)
187         if key == 32:
188             mode = next(modes)
189             if mode == "depth":
190                 cv2.imshow('Image', img_depth)
191             if mode == "color":
192                 if img_color.size:
193                     cv2.imshow('Image', img_color)
194
195     if key == 27: # Esc key to stop
196         break

```

## **A.0.2 Chamfer Script**

```

1  import numpy as np
2  from scipy.spatial import distance
3
4  # Define two sets of points (sensor astra line 6 -> set_A and kinect line 37 -> set_B)
5  set_A_100 = np.array([
6      [100.6367645, 593.5106201, 2178.698975],
7      [125.999794, 488.879364, 2114.458496],
8      [104.2917252, 149.6486511, 2094.320313],
9      [107.4500275, -54.99118042, 2072.356934],
10     [146.1627197, 403.2581177, 2291.689209],
11     [-10.78676605, 393.1947327, 2220.676025],
12     [146.1627197, 111.6621704, 2291.689209],
13     [131.4588623, 311.244873, 1979.680908],
14     [146.1627197, -155.3651123, 2291.689209],
15     [340.797699, 228.7185059, 1835.909546],
16     [201.2516174, -168.0711212, 2019.991577],
17     [32.87856293, -184.5029755, 2120.921631],
18     [128.497345, -566.2248535, 1831.840332],
19     [-1.257781029, -619.5349121, 2027.10022],
20     [137.8198853, -975.1410522, 1805.037476],
21     [82.2947464, -1012.133789, 2110.184082],
22     [146.1627197, -264.194458, 2291.689209],
23     [382.6654663, 212.2132263, 1807.155273],
24     [120.401001, 401.3873901, 2109.264648],
25     [120.401001, 401.3873901, 2109.264648]
26 ])
27
28 set_B_100 = np.array([
29     [100.6719666, 871.5637817, 2159.276367],
30     [77.84082031, 763.2202148, 2117.588623],
31     [87.68508148, 435.9494934, 2079.401855],
32     [91.98124695, 233.7402802, 2048.213623],
33     [115.6095886, 636.3069458, 2247.461182],
34     [49.1781311, 661.7231445, 1965.020142],
35     [115.6095886, 354.3705444, 2247.461182],
36     [85.76293945, 469.9833984, 1761.585815],
37     [115.6095886, 81.47161865, 2247.461182],
38     [195.7979126, 376.3834229, 1530.057861],
39     [198.923645, 93.07858276, 2053.357178],
40     [-14.00746155, 120.2686691, 1984.830933],
41     [108.2102737, -306.2295837, 1885.459961],
42     [110.1070633, -290.3496094, 1875.956665],
43     [108.2102737, -732.1087036, 1885.459961],
44     [207.7293396, -703.4514771, 1841.463623],
45     [115.6095886, -23.68541718, 2247.461182],
46     [217.8049011, 357.6634216, 1483.752319],
47     [80.37509918, 678.9685669, 2107.757813],
48     [80.37509918, 678.9685669, 2107.757813]
49 ])
50
51 set_A_0 = np.array([
52     [0.3780231476, 564.6588135, 2077.658936],
53     [-25.22777939, 457.964325, 2129.618896],
54     [-42.78187561, 133.2587585, 2118.761719],
55     [-47.87962341, -65.11546326, 2092.872803],
56     [114.3416443, 366.4434204, 2176.645264],
57     [-170.2688446, 380.2510376, 2074.929688],
58     [238.800766, 143.8475952, 2285.530518],
59     [-327.8029785, 184.3431091, 1957.890625],
60     [382.8813477, 31.1997261, 2460.561523],
61     [-533.6550903, 67.07290649, 1868.687012],
62     [48.80004883, -164.3721619, 2103.413818],
63     [-149.2009735, -160.0801392, 2071.936279],
64     [44.10560226, -560.9232788, 1978.891479],
65     [-151.1220093, -529.1920776, 1880.799194],
66     [54.94152832, -950.225708, 1962.840088],
67     [-151.1220093, -918.9759521, 1880.799194],
68     [441.7904968, -14.85776138, 2532.125244],
69     [-574.8255005, 43.61886597, 1850.846313],
70     [-29.73426056, 374.6060486, 2126.831543],
71     [-29.73426056, 374.6060486, 2126.831543]
72 ])

```