

Framework para o Desenvolvimento de Aplicações de Treinamento da Operação de Máquinas Utilizando Realidade Aumentada

Vinicius Chrisosthemos Teixeira
School of Technology
Porto Alegre, Brazil
vinicius.teixeira.002@acad.pucrs.br

Marcio Sarroglia Pinho (orientador)
School of Technology
Porto Alegre, Brazil
pinho@pucrs.br

1 INTRODUÇÃO

A Realidade Aumentada (RA) é uma tecnologia que permite a inserção de elementos 3D na exibição de ambientes reais através de uma câmera. A Figura 1 apresenta um exemplo de aplicação de Realidade Aumentada, onde um modelo 3D simples de uma seta é exibido em cima de um objeto real.

No caso de aplicações de RA para o treinamento de operações e procedimentos, o usuário pode utilizar um capacete ou uma câmera que capta as imagens do mundo real e sobre estas imagens inserem-se elementos 3D como uma chave de fenda ou um alicate no campo de visão do usuário, podendo passar instruções de qual ferramenta utilizar ou onde usá-la.



FIGURA 1 - EXEMPLO DE APLICAÇÃO UTILIZANDO REALIDADE AUMENTADA
Fonte: O Autor

As aplicações de RA voltadas à educação ganharam visibilidade nos últimos anos devido ao fato de suas características proporcionarem um desempenho igual ou superior no processo de aprendizagem em comparação com os métodos tradicionais, como mostram alguns estudos na

área (Barsom et al., 2016; Heinz et al., 2019; Munzer et al., 2019; Zhu et al., 2015). Na Medicina, por exemplo, aplicações voltadas ao aprendizado da anatomia humana, como a visualização 3D de imagens de tomografias computadorizadas de partes específicas do corpo, como tronco e cérebro, possuem efetividade na educação equivalentes às formas tradicionais de ensino e uma melhora significativa no interesse e participação dos envolvidos, em especial se for possível sobrepor a imagem ao corpo do próprio paciente (Ma et al., 2016; Moro et al., 2017; Thomas et al., 2010).

Na área da Indústria, por sua vez, encontram-se vários projetos, tanto acadêmicos quanto profissionais, voltados à capacitação para o manuseio de máquinas (Cao et al., 2020; Heinz et al., 2019). Esses projetos, em geral, possuem um modelo 3D da máquina com outros elementos gráficos como imagem, vídeo ou animações para auxiliar o operador no treinamento ou manutenção, gerando um ganho na efetividade e acurácia da tarefa junto com uma maior compreensão do processo pelo operador (Bottani and Vignali, 2019).

Analisando as aplicações de RA voltadas ao treinamento, notam-se algumas similaridades entre elas, em especial a existência de etapas que devem ser cumpridas, a utilização de ferramentas e a necessidade de acionamento de chaves e botões. Apesar destas similaridades, existem poucas ferramentas que deem suporte a este desenvolvimento. Ferramentas como Unreal (“Unreal Engine,” 2022) ou Unity3D (Unity, 2022) são excelentes,

mas demandam a criação de aplicações a partir do zero, requerendo um maior esforço de desenvolvimento.

Em face desta lacuna, este trabalho propõe um framework de desenvolvimento de aplicações de RA voltadas a processos de treinamento da operação de máquinas. A Plataforma-alvo do framework é o capacete Oculus Quest 2, exibido na Figura 2.



FIGURA 2 - OCULUS QUEST 2 UTILIZADO NO TRABALHO
Fonte: O Autor

O framework foi baseado nas características observadas em aplicações de treinamento na indústria, envolvendo máquinas complexas (Blattgerste et al., 2017; Cao et al., 2020; Gavish et al., 2015) e segue alguns padrões das ferramentas já existentes no mercado. As características essenciais observadas nos estudos foram:

- **Exibição de instruções ou instrumentos como elementos gráficos ou 3D:** Característica essencial que permite uma maior compreensão da tarefa e agilidade na operação;
- **Rastreamento da posição do usuário:** Permitindo que o usuário se posicione em locais cujo o ângulo de visão favoreça a compreensão da instrução exibida;
- **Suporte à definição de etapas:** Para guiar o usuário ao final do procedimento resolvendo um problema de cada vez, permitindo avançar/retroceder.

No que diz respeito ao processo de desenvolvimento, é possível especificar todas as etapas de treinamento em um arquivo texto como um script. Entretanto, a ideia central do framework é permitir que todo o processo de especificação do treinamento seja realizado no mesmo ambiente imersivo, ou aumentado, em que ocorre o treinamento. Neste ambiente, é possível:

- **Instanciar e manipular objetos 3D:** Para permitir a edição do ambiente de treinamento de forma fácil, intuitiva e com maior precisão;
- **Visualizar os objetos 3D com Oclusão:** Para auxiliar o usuário na percepção de espaço e no posicionamento dos objetos 3D;
- **Rastrear a posição do usuário:** Para permitir que o usuário explore o cenário de treinamento de vários ângulos diferentes e interaja no ambiente com movimentos naturais;
- **Definir de uma sequência de exibições de objetos:** Permitindo ao usuário montar uma sequência de etapas com objetos específicos para uma determinada tarefa.

2 OBJETIVOS

2.1 Objetivo Geral

O objetivo deste trabalho é a construção de um framework de desenvolvimento de aplicações de RA voltadas ao treinamento de operação de máquinas. O framework permite que se definam **etapas** de um treinamento que ocorre em um ambiente de realidade aumentada. Em cada etapa podem ser inseridas instruções textuais, vídeos, e objetos 3D estáticos ou animados.

2.2 Objetivos Específicos

Como objetivos específicos tem-se:

- Pesquisar ferramentas de criação de aplicações de RA;
- Desenvolvimento de um framework de RA que permita a criação de uma sequência de exibição de objetos e a manipulação deles;
- Criar uma aplicação como estudo de caso com o objetivo de testar o poder expressivo do framework na criação de treinamento envolvendo uma máquina real;
- Discutir os ganhos e limitações do framework desenvolvido.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Realidade Aumentada

Segundo Schmalstieg (Schmalstieg and Höllerer, 2016), RA é uma tecnologia que combina a visão de um usuário do mundo real com imagens digitais, com o objetivo de aprimorar o entendimento do usuário sobre este mundo. Para que seja possível obter um sistema de RA realmente útil, as imagens digitais devem estar registradas ao mundo

real de forma adequada, respeitando a posição dos objetos reais, bem como eventuais oclusões geradas pelo posicionamento das imagens no entorno de objetos reais, além de permitir interação em tempo real por parte do usuário (Azuma, 1997).

Na área específica de treinamento, a RA pode facilitar a maior parte do processo de aprendizagem exibindo informações úteis no campo de visão do usuário conforme ele vai avançando nas etapas do processo. Para tanto, podem ser exibidas as peças ou ferramentas nos locais onde devem ser encaixadas ou retiradas, além de animações que demonstrem como realizar os procedimentos etc.

4 TRABALHOS RELACIONADOS

Dentro da pesquisa realizada na literatura envolvendo frameworks de criação de aplicações voltadas ao treinamento, não foram encontrados projetos que se encaixassem no modelo estabelecido para este trabalho. Os trabalhos citados dessa seção são voltados à utilização de aplicações simples de RA, permitindo funcionalidades básicas como instanciar e manipular objetos virtuais e salvar suas posições entre seções.

Com o objetivo de permitir a criação de aplicações de RA, sem o uso de programação, o Adobe Aero (Adobe, 2022) permite o instanciamento de objetos virtuais em um ambiente real utilizando tanto os objetos disponibilizados pela ferramenta quanto carregados ou customizados pelo usuário. Para o registro dos objetos no mundo real a ferramenta suporta a utilização de marcadores fiduciais e o escaneamento de superfícies planas sobre a qual se pode criar objetos. É possível ainda definir comportamento para os objetos como giros e descolamentos. Pode-se ainda definir gatilhos/eventos para que os movimentos ocorram como proximidade entre o objeto e usuário ou a colisão entre objetos virtuais. A ferramenta, que roda em celulares, permite também a edição de cenários em uma aplicação para computador. A partir desta plataforma, é possível gravar o cenário em nuvem e a utilizá-lo posteriormente em celulares. A Figura 3 apresenta um exemplo de aplicação no Adobe Aero. A ferramenta não requer nenhuma

codificação por parte do desenvolvedor, tornando-se mais acessível para o público geral.

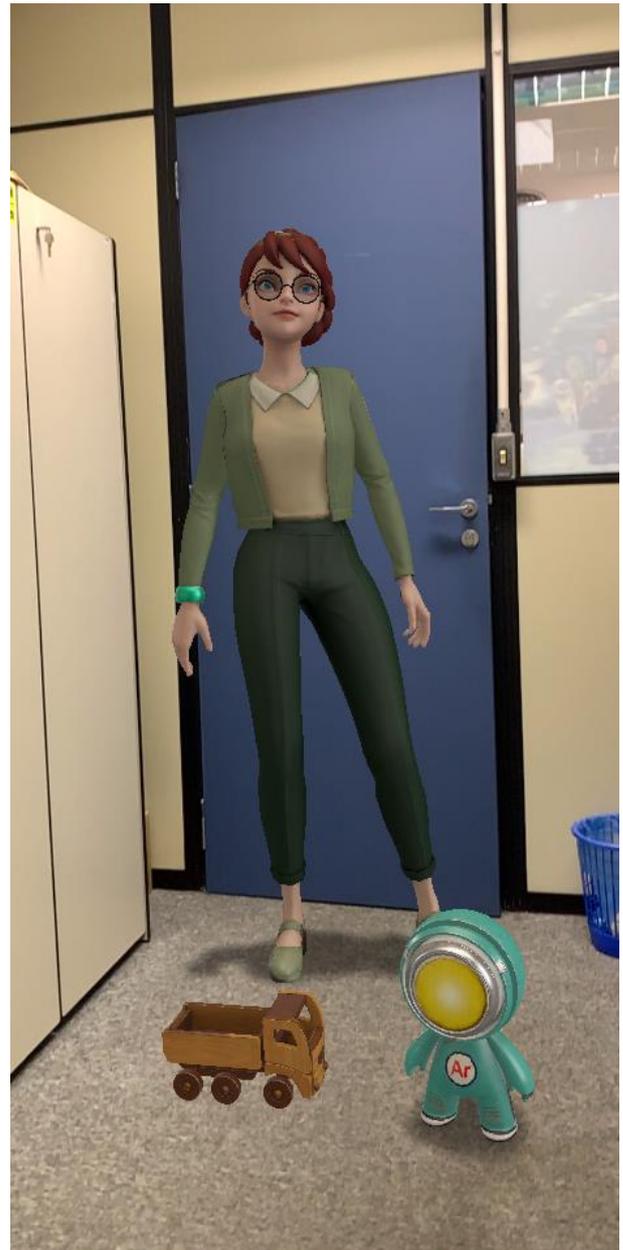


FIGURA 3 - EXEMPLO DE OBJETO 3D INSTANCIADO NO ADOBE AERO
Fonte: O Autor

Aprimorando esse conceito, ferramentas como a Librestream (“LibreStream,” 2022), cujo exemplo de aplicação é apresentado na Figura 4, são utilizadas para assistência remota, mas, também pode ser usada para o treinamento individual. Nele, o operador utiliza um dispositivo móvel que filma um cenário e envia as imagens, em tempo real, para um especialista que, remotamente, o ajuda na tarefa em questão.

5 FRAMEWORK

Para dar suporte à construção de aplicações de treinamento de máquinas utilizando Realidade Aumentada, o framework desenvolvido disponibiliza uma série de funcionalidades, na forma de scripts da Unity3D, capazes de manipular um conjunto de objetos e criar uma aplicação de treinamento, que roda em um capacete de realidade virtual/aumentada.

O framework suporta dois modos de iteração, o **Modo de Edição** e um **Modo de Treinamento**. No **Modo de Treinamento**, o usuário possui apenas acesso ao painel de controle e pode navegar pelas etapas e trocar para o modo de Edição. Já o **Modo de Edição**, o usuário ganha acesso a vários mecanismos para estruturar um treinamento como a criação das **Etapas de Treinamento** dentro das quais são inseridos **Elementos de Interação**. Para inserir os Elementos de Interação, existe uma **Biblioteca de Objetos** pré-definida a partir da qual os são instanciados no cenário virtual de cada etapa.

Para navegar entre as etapas do treinamento, o usuário dispõe de um **Painel de Controle**. Este Painel está disponível tanto no modo de Edição, quanto de Treinamento. Neste segundo modo, destinado ao usuário que precisa ser treinado, o Painel de Controle permite avançar e retroceder nas etapas do treinamento e executar as ações planejadas.

A Figura 5 apresenta o diagrama de classes resumido do framework. Nele, temos a classe **ApplicationManager** responsável por gerenciar os estados da aplicação, alternando em modo de Treinamento e de Edição, além de outras funções como desligar a visualização objeto referência e de iniciar o processo de ancoragem.

O fluxo de execução do programa é orientado a eventos, sendo a classe **InputManager**, um Singleton visível para todas as classes do sistema, responsável por encapsular os eventos de entrada de dados do usuário, como ações do joystick e a entrada de texto utilizando o teclado virtual. No início da aplicação, as classes vinculam métodos a serem executados para cada botão pressionado pelo usuário. Assim, cada entidade do programa, como o Painel de

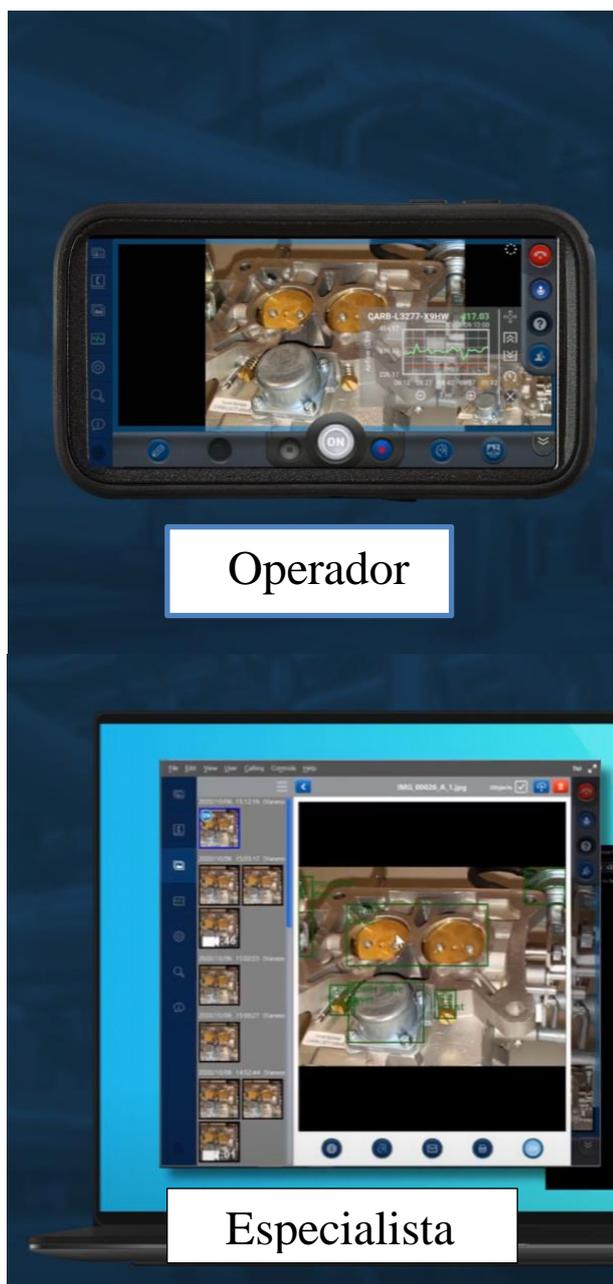


FIGURA 4 - EXEMPLO DE APLICAÇÃO UTILIZANDO LIBRESTREAM
Fonte: ("LibreStream," 2022)

Outros projetos focam em desenvolver sistemas de treinamento específicos para suas áreas, como o treinamento de protocolos de saúde (Barsom et al., 2016), manutenção de máquinas complexas na indústria (Cao et al., 2020) ou o aprendizado de algum tema na educação (Wu et al., 2013). Essas ferramentas compartilham características semelhantes como a utilização de um modelo 3D representando o objeto de interesse, o rastreamento da posição do usuário para visualizar as informações inseridas na aplicação e suporte a dados multimídia, como texto, imagem e vídeos e outros.

Controle, recebe a ação disparada pela InputManager, como um clique de botão da interface ou a movimentação de algum objeto na cena, e trata esses eventos de acordo com o método vinculado a ele. O fluxo de execução do Framework é apresentado na Figura 6.

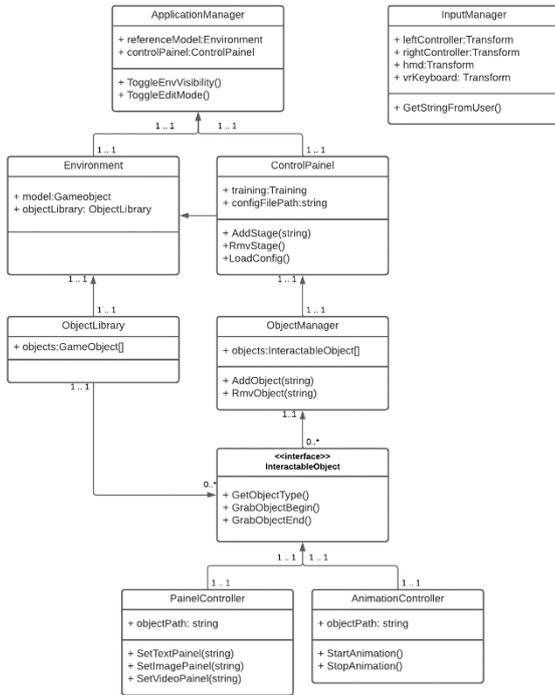


FIGURA 5 - DIAGRAMA UML DO FRAMEWORK
Fonte: O Autor

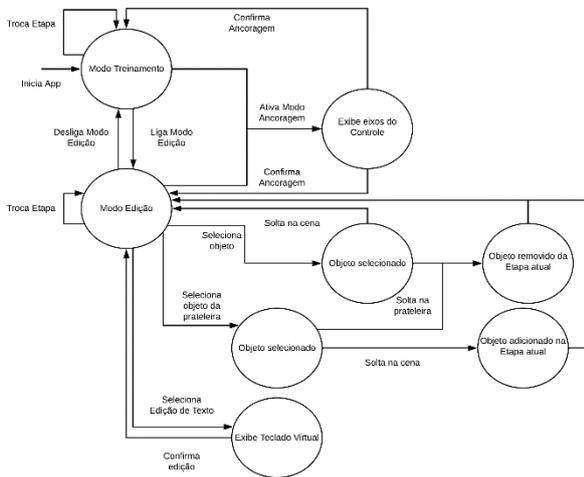


FIGURA 6 - MÁQUINA DE ESTADOS DO FRAMEWORK
Fonte: O Autor

O restante deste capítulo apresenta a funcionalidades de ancoragem entre os ambientes real e virtual, os de elementos de interação e a descrição das formas de interação 3D suportadas pelo Framework.

5.1 Ancoragem/Registro entre os Mundos Real e Virtual

Para permitir que os objetos virtuais posicionados durante a criação do treinamento apareçam nos locais corretos, em relação ao mundo real, na aplicação que roda no capacete, é necessário realizar o **registro do mundo virtual no mundo real**. Para tanto, ao iniciar a aplicação, o usuário indica uma posição de referência no mundo real. Esta posição, representa também a origem do sistema de coordenadas do ambiente virtual, a partir da qual todos os demais objetos virtuais são posicionados.

Para indicar esta posição, o usuário apoia o joystick ou o dedo indicador da mão direita sobre Ponto de Ancoragem. Um exemplo do momento desta ancoragem pode ser visto na Figura 7.

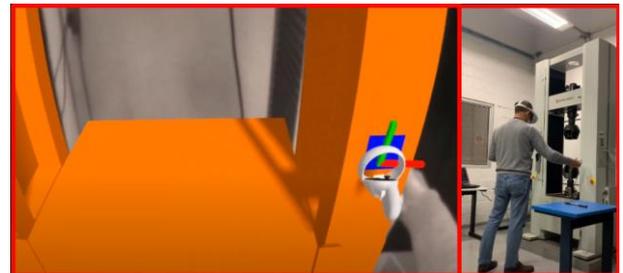


FIGURA 7 - MOMENTO DA ANCORAGEM DO AMBIENTE VIRTUAL UTILIZANDO UM JOYSTICK
Fonte: O Autor

5.2 Elementos de Interação

Os elementos manipulados pela aplicação devem ser especificados em um arquivo texto no formato JSON que contém uma Descrição/Título do treinamento, um **Modelo de Objeto de Referência**, a descrição da **Biblioteca de Objetos**, a posição inicial do **Painel de Controle** e uma lista de **Etapas**. Dentro de cada Etapa, são instanciados um conjunto de **Instâncias de Etapa**. Um exemplo de descrição de um treinamento é exibido na Figura 8.

5.2.1 Modelo de Referência

O **Modelo Referência** representa os objetos do ambiente real e é usado para suportar oclusão dos objetos virtuais com os objetos reais. Também é usado como referência de posicionamento para os demais objetos virtuais. O Modelo é único e está disponível em todas as etapas do treinamento.

Para suportar o efeito de oclusão, o Modelo de Referência é renderizado com um material especial transparente que não permite a renderização de objetos virtuais posicionados atrás dele em relação ao usuário. A

Figura 9 mostra um exemplo do tratamento de oclusão suportado pela aplicação.

```
1  {
2  "name": "Tutorial 1",
3  "mainModel": "MaquinaEnsaioUniversal/ModeloReferencia",
4  "painelPosition": { "x": 0.0, "y": 0.0, "z": 0.0 },
5  "painelRotation": { "x": 0.0, "y": 0.0, "z": 0.0 },
6  "libraryPosition": { "x": 0.0, "y": 0.0, "z": 3.0 }
7  "libraryRotation": { "x": 0.0, "y": 180.0, "z": 0.0 }
8  "stages": [
9    {
10     "stageElements": [
11       {
12         "position": { "x": 1, "y": 1, "z": -0.06300000101327896 },
13         "eulerRotation": { "x": 0.0, "y": 0.0, "z": 0.0 },
14         "scale": { "x": 1.0, "y": 1.0, "z": 1.0 },
15         "assetName": "MaquinaEnsaioUniversal/Treinamento_1/Texto_inicial.txt",
16         "objType": 0,
17         "occlusion": false
18       },
19       {
20         "position": { "x": 1, "y": 0.9120000004768372, "z": 0 },
21         "eulerRotation": { "x": 0.0, "y": 0.0, "z": 0.0 },
22         "scale": { "x": 1.0, "y": 1.0, "z": 1.0 },
23         "assetName": "MaquinaEnsaioUniversal/Treinamento_1/Animacao_Subindo_Encaixe",
24         "objType": 2,
25         "occlusion": true
26       },
27       {
28         "position": { "x": 0, "y": 2, "z": 1 },
29         "eulerRotation": { "x": 0.0, "y": 0.0, "z": 0.0 },
30         "scale": { "x": 1.0, "y": 1.0, "z": 1.0 },
31         "assetName": "MaquinaEnsaioUniversal/Treinamento_1/Seta",
32         "objType": 1,
33         "occlusion": true
34       }
35     ],
36     "description": "Etapa 1",
37     "finished": false
38   }
39 ],
40 "currentStage": 0
41 }
```

FIGURA 8 - EXEMPLO DE UM TREINAMENTO DESCRITO EM FORMATO JSON
Fonte: O Autor



FIGURA 9 – TRATAMENTO DE OCLUSÃO COM UM OBJETO DE REFERÊNCIA
Fonte: O Autor

5.2.2 Painel de Controle

O **Painel de Controle**, que pode ser visto na Figura 10, é um objeto especial voltado à interação do usuário com as etapas do treinamento. Nele, o usuário pode escolher entre os modos de utilização do ambiente, apontando a engrenagem no canto superior esquerdo do painel selecionando o checkbox “Edit Mode”.

No **Modo de Treinamento**, é possível navegar pelas tarefas existentes e sinalizar quais foram completadas com o checkbox “Done”, seguindo o fluxo de tarefas definido. Também é possível mover o painel de controle colocando-o em uma posição escolhida pelo usuário.

Além destas funcionalidades, no **Modo de Edição**, o usuário ganha liberdade para inserir, remover e editar as **Instâncias de Etapa**, além de poder adicionar ou remover etapas, usando os botões “+ Task” e “- Task”. As posições das instâncias e a configuração de etapas são sempre salvas no arquivo de configuração.

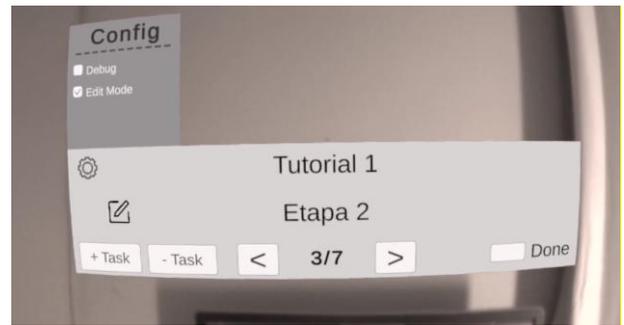


FIGURA 10 - PAINEL DE CONTROLE
Fonte: O Autor

5.2.3 Biblioteca de Objetos

A **Biblioteca de Objetos** define um conjunto de objetos disponíveis para serem inserido na Etapa atual. Esses objetos são descritos em um segundo arquivo de configuração, contendo o caminho para o objeto e seus atributos de instanciamento (posição, rotação e escala). Na versão atual do framework, todos os objetos da biblioteca devem ser compilados junto com a aplicação, mas, em uma versão futura, se pode incluir uma funcionalidade de carga destes objetos a partir de um servidor. Visualmente, a biblioteca é renderizada como uma espécie de prateleira, envolta em um paralelepípedo semitransparente, como na Figura 11. A posição da Biblioteca é definida no arquivo JSON.

A criação de um novo objeto é feita “agarrando-se” o objeto e removendo-o da prateleira. Esse processo adiciona uma nova instância do objeto. Para remover um objeto de uma etapa, basta agarrar o objeto e colocá-lo novamente na prateleira. Cabe ressaltar que toda a edição realizada através deste processo é gravada no arquivo de configuração, podendo ser usada posteriormente.

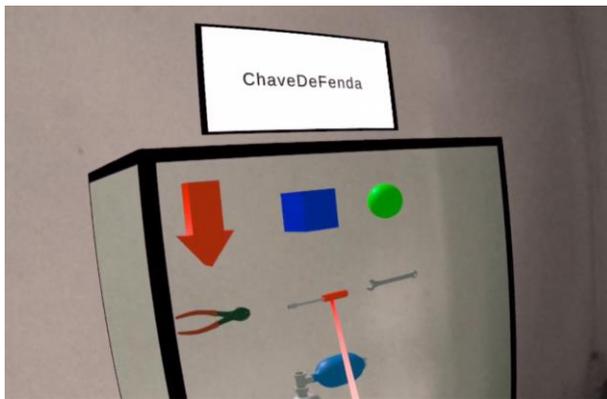


FIGURA 11 - BIBLIOTECA DE OBJETOS DENTRO DA APLICAÇÃO
Fonte: O Autor

5.2.4 Instâncias de Etapa

Conforme já mencionado, um Treinamento é definido por uma lista de **Etapas**. Cada Etapa é definida por uma descrição, um valor booleano sinalizando se ela foi completada ou não, e uma lista de **Instâncias de Etapa**. Uma instância de Etapa é composta por um identificador do seu modelo, com base nos objetos disponíveis na biblioteca, um identificador do tipo de objeto, podendo ser um **Modelo 3D**, **Painel de Texto** ou **Exibição Gráfica** e uma **Animação**, um valor booleano sinalizando se o objeto possui oclusão ou não, e sua posição, rotação e escala iniciais.

As subseções a seguir descrevem estes tipos de objetos.

5.2.4.1 Modelos 3D

Objetos do tipo **Modelos 3D** são modelados e inseridos como "prefabs" da Unity. Estes modelos possuem apenas os scripts necessários para serem selecionados e movimentados pelo usuário. A Figura 12 apresenta um exemplo de uma "seta" instanciada sobre uma cena de realidade aumentada.

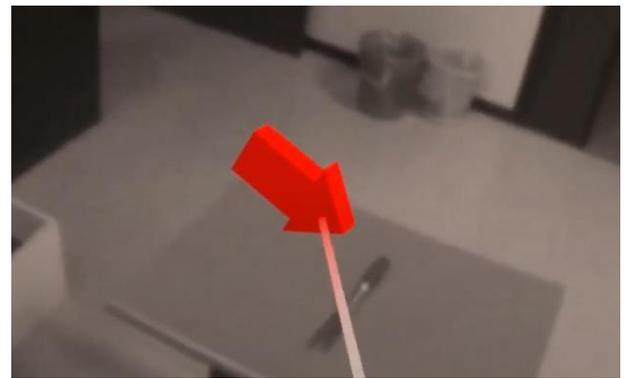


FIGURA 12 - EXEMPLO DE MODELO 3D
Fonte: O Autor

Para especificar o instanciamento de um Modelo 3D, o script JSON deve conter o formato descrito entre as linhas 28 e 33, na Figura 8.

A fim de dar suporte à oclusão de objetos virtual por objetos reais, o script JSON pode conter objetos 3D em que a TAG "*occlusion*" é definida com o valor "true". Com isto, a geometria do objeto é associada a um script de renderização específico que implementa a oclusão.

Outro aspecto importante a ressaltar é que, ao instanciar os objetos 3D no ambiente virtual, o software posiciona estes objetos a partir do **Ponto de Ancoragem** (seção 5.1) do ambiente real.

5.2.4.2 Painel de Exibição

Para permitir a exibição de mensagens ao usuário o framework possui uma abstração chamada **Painel**, que permite instanciar painéis na forma de textos, imagens e vídeos. O componente responsável por instanciar os objetos identifica qual Painel deve ser instanciado com base na extensão do arquivo informado. Extensões do tipo *.txt* são instanciados como Painéis de Texto e extensões do tipo *.png*, *.jpg* e *.mp4* como Painéis de Exibição Gráfica.

Os **Painéis de Texto** são utilizados para a exibição de textos com o objetivo de fornecer instruções ou dados ao usuário. A Figura 13 apresenta um exemplo de instruções exibidas textualmente ao usuário. Para especificar o instanciamento de um Painel de Texto e Exibição Gráfica, o script JSON deve conter o formato descrito entre as linhas 12 e 17, na Figura 8.

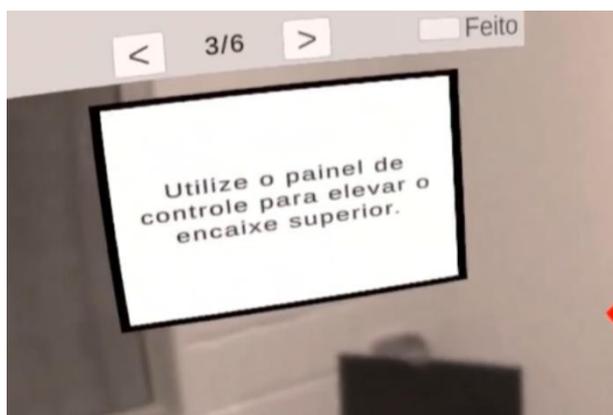


FIGURA 13 - PAINEL DE TEXTO
Fonte: O Autor

O **Painel de Exibição Gráfica**, por sua vez, permite utilizar imagens ou vídeos demonstrando etapas de um procedimento, diagramas de algum painel de controle, etc. No caso do uso de vídeos, o usuário tem a possibilidade de iniciar, pausar ou repetir a exibição. A Figura 14 apresenta um exemplo de um vídeo apresentado ao usuário.

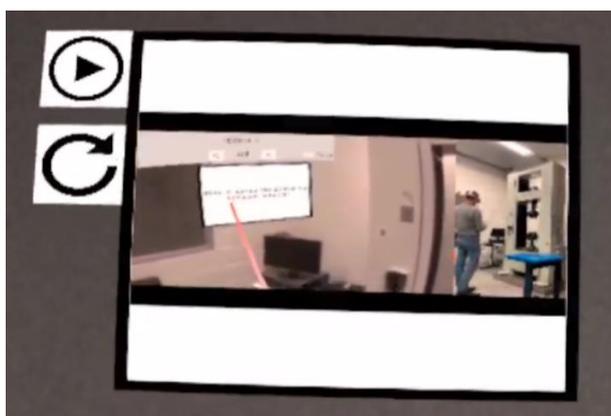


FIGURA 14 - PAINEL DE EXIBIÇÃO GRÁFICA - VÍDEO
Fonte: O Autor

Por fim, os objetos do tipo **Animação** podem executar qualquer tipo de animação 3D suportada pela Unity. A animação é iniciada assim que o objeto é instanciado no ambiente virtual. Para especificar o instanciamento de uma Animação, o script JSON deve conter o formato descrito entre as linhas 20 e 25, na Figura 8.

5.3 Entradas do Usuário

Para permitir que o usuário modifique o ambiente virtual do treinamento foi desenvolvido um script para o framework que centraliza todos os eventos de entradas do joystick. Este script permite a seleção e a manipulação de objetos 3D e a entrada de textos.

Para a seleção e manipulação de objetos, o script implementa o método Raycast (Bolt, 1980) gerando um raio que parte da ponta do joystick, com mesma orientação deste joystick. A técnica permite apontar o primeiro objeto tocado pelo raio. Feito o apontamento, ao pressionar e segurar o gatilho do joystick, o usuário pode mover as Instâncias de Etapa como desejar. Do ponto de vista da implementação, esse processo se resume em definir o objeto como filho do joystick, que terá sua posição alterada de acordo com a movimentação aplicada no joystick.

Para a entrada de textos foi criado um Teclado Virtual baseado no trabalho de RealaryVR (RealaryVR, 2022). Este teclado é composto um modelo 3D de um teclado e um **apontador** conectado ao joystick. A Figura 15 mostra estes dois elementos. Para acionar uma tecla, o usuário deve tocá-la com o apontador. Para a implementação, cada tecla possui um colisor que dispara um evento de inserção de um caracter no campo de texto. Ao finalizar a edição, o usuário seleciona a tecla “Confirm” e o texto é enviado para o componente que está sendo editado.



FIGURA 15 - TECLADO VIRTUAL
Fonte: O Autor

6 ESTUDO DE CASO

A fim de demonstrar as possibilidades do framework desenvolvido, foi criado um treinamento de manuseio de uma Máquina de Ensaio Universal como estudo de caso. Esta máquina, que aparece no lado direito da Figura 7, faz parte do acervo de equipamentos do Laboratório de Materiais (LAMAT), da Escola Politécnica.

Este treinamento possui um modelo 3D da máquina-alvo junto com alguns objetos primitivos, como setas, cubos e esferas, e alguns objetos mais complexos, como chave de

fenda e alicate. O objetivo do treinamento é **encaixar um corpo de prova no local no local correto da máquina**, seguindo as orientações criadas pelo instrutor.

O treinamento é composto por 6 etapas e utiliza uma lista predefinida de objetos relacionados com o procedimento. A Tabela 1 descreve cada etapa com seus respectivos objetos.

TABELA 1 - DESCRIÇÃO DAS TAREFAS DO TUTORIAL

Etapa	Descrição	Objetos
1	Introdução ao tutorial	- Painel de Texto
2	Iniciar a Máquina de Ensaio Universal	- Painel de Texto - Seta (Modelo Simples)
3	Elevar o encaixe superior	- Painel de Texto - Seta (Modelo Simples) - Animação do encaixe superior subindo
4	Encaixar o corpo de prova no encaixe inferior	- Painel de Texto - 2 Setas
5	Descer o encaixe superior até o corpo de prova	- Painel de Texto - Seta - Animação do encaixe superior descendo
6	Tutorial finalizado	- Painel de Texto

O arquivo de configuração foi descrito com 6 etapas, tendo painéis de texto específicos para cada etapa junto com apontadores em locais arbitrários. As posições desses objetos foram alteradas pelo instrutor no modo de edição imersiva disponível no Painel de Controle. O resultado é exibido no Apêndice 1 e no vídeo de demonstração¹.

7 CONCLUSÃO

Neste trabalho desenvolvemos um framework para a construção de aplicações de treinamento com Realidade Aumentada. A ferramenta possibilita a criação de uma sequência de etapas de visualização de objetos virtuais em locais escolhidos pelo instrutor em uma ambiente real, de forma imersiva e a posterior interação do usuário com estas etapas..

Para testar a ferramenta, criou-se um estudo de caso no qual um tutorial foi desenvolvido para manusear uma Máquina de Ensaio Universal. As etapas do tutorial foram descritas e traduzidas para um arquivo de configuração inicial que foi posteriormente alterado, no editor imersivo. No geral, a aplicação final demonstrou um bom desempenho ao guiar o usuário pelas etapas estabelecidas pelo instrutor e na compreensão da tarefa como um todo.

8 TRABALHOS FUTUROS

O framework mostrou-se genérico o suficiente para a construção de aplicações simples de treinamento, porém ainda mantém um vínculo forte com o editor da Unity ou o próprio arquivo de configuração. A fim de ganhar mais autonomia dentro da aplicação, a criação de um mecanismo de importação de objetos via internet se tornará indispensável para versões futuras, permitindo que o instrutor invista mais tempo dentro da aplicação, organizando as tarefas e refinando o posicionamento dos objetos, do que compilando uma nova aplicação com novos objetos.

Seguindo essa linha de tornar a aplicação ainda mais genérica, a criação de “sessões” dentro da aplicação permitiria a um instrutor criar diferentes tutoriais e compartilhá-los com o mesmo mecanismo de importação de objetos citados anteriormente. Estas sessões permitiram o instrutor escolher a biblioteca de objetos a ser utilizada na sessão e exportá-la para um servidor, onde seria listada na pesquisa por tutoriais do usuário.

A atenção para a experiência do usuário também deve estar na prioridade para os próximos passos. Utilizar sons para auxiliar a compreensão de ações como selecionar, agarrar e clicar pode tornar a aplicação mais agradável e imersiva. Outro detalhe importante é a utilização dos controles (joystick) pelo usuário, que limita suas ações durante o treinamento e pode interferir negativamente em seu desempenho, tornando assim a utilização de mecanismo

¹ <https://www.youtube.com/watch?v=Jcgqq2510Ww>

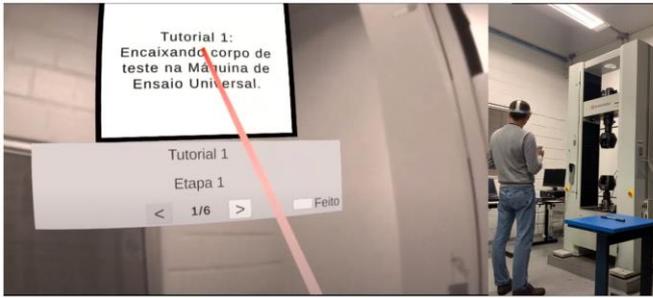
de rastreamento de mãos um ponto importante para novas versões da ferramenta.

REFERÊNCIAS

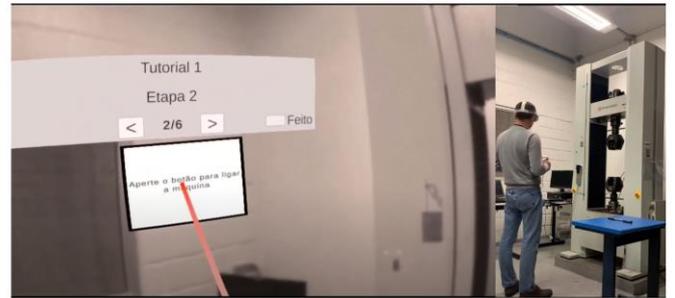
- Adobe, 2022. Adobe Aero [WWW Document]. URL <https://www.adobe.com/br/products/aero.html> (accessed 6.4.22).
- Barsom, E.Z., Graafland, M., Schijven, M.P., 2016. Systematic review on the effectiveness of augmented reality applications in medical training. *Surg Endosc* 30, 4174–4183. <https://doi.org/10.1007/s00464-016-4800-6>
- Blattgerste, J., Strenge, B., Renner, P., Pfeiffer, T., Essig, K., 2017. Comparing Conventional and Augmented Reality Instructions for Manual Assembly Tasks, in: Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments. Presented at the PETRA '17: 10th International Conference on Pervasive Technologies Related to Assistive Environments, ACM, Island of Rhodes Greece, pp. 75–82. <https://doi.org/10.1145/3056540.3056547>
- Blender 3D [WWW Document], 2022. URL <https://www.blender.org/> (accessed 6.4.22).
- Bottani, E., Vignali, G., 2019. Augmented reality technology in the manufacturing industry: A review of the last decade. *IISE Transactions* 51, 284–310. <https://doi.org/10.1080/24725854.2018.1493244>
- Cao, Y., Qian, X., Wang, T., Lee, R., Huo, K., Ramani, K., 2020. An Exploratory Study of Augmented Reality Presence for Tutoring Machine Tasks, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. Presented at the CHI '20: CHI Conference on Human Factors in Computing Systems, ACM, Honolulu HI USA, pp. 1–13. <https://doi.org/10.1145/3313831.3376688>
- Demo, 2022. Demonstração de Ambiente de Treinamento com Realidade Aumentada.
- Gavish, N., Gutiérrez, T., Webel, S., Rodríguez, J., Peveri, M., Bockholt, U., Tecchia, F., 2015. Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks. *Interactive Learning Environments* 23, 778–798. <https://doi.org/10.1080/10494820.2013.815221>
- Heinz, M., Büttner, S., Röcker, C., 2019. Exploring training modes for industrial augmented reality learning, in: Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments. Presented at the PETRA '19: The 12th Pervasive Technologies Related to Assistive Environments Conference, ACM, Rhodes Greece, pp. 398–401. <https://doi.org/10.1145/3316782.3322753>
- LibreStream [WWW Document], 2022. URL <https://librestream.com/use-cases/> (accessed 6.4.22).
- Ma, M., Fallavollita, P., Seelbach, I., Von Der Heide, A.M., Euler, E., Waschke, J., Navab, N., 2016. Personalized augmented reality for anatomy education: Personalized Augmented Reality. *Clin. Anat.* 29, 446–453. <https://doi.org/10.1002/ca.22675>
- Moro, C., Stromberga, Z., Raikos, A., Stirling, A., 2017. The effectiveness of virtual and augmented reality in health sciences and medical anatomy: VR and AR in Health Sciences and Medical Anatomy. *American Association of Anatomists* 10, 549–559. <https://doi.org/10.1002/ase.1696>
- Munzer, B.W., Khan, M.M., Shipman, B., Mahajan, P., 2019. Augmented Reality in Emergency Medicine: A Scoping Review. *J Med Internet Res* 21, e12368. <https://doi.org/10.2196/12368>
- RealVR, 2022. VR Keyboard.
- Schmalstieg, D., Höllerer, T., 2016. Augmented Reality: Principles and Practice 2.
- Thomas, R.G., William John, N., Delieu, J.M., 2010. Augmented Reality for Anatomical Education. *Journal of Visual Communication in Medicine* 33, 6–15. <https://doi.org/10.3109/17453050903557359>
- Unity, U., 2022. Unity Engine [WWW Document]. URL <https://unity.com/> (accessed 6.10.22).
- Unreal Engine [WWW Document], 2022. Unreal Engine. URL <https://www.unrealengine.com/en-US> (accessed 6.10.22).
- Wu, H.-K., Lee, S.W.-Y., Chang, H.-Y., Liang, J.-C., 2013. Current status, opportunities and challenges of augmented reality in education. *Computers & Education* 62, 41–49. <https://doi.org/10.1016/j.compedu.2012.10.024>
- Zhu, E., Lilienthal, A., Shluzas, L.A., Masiello, I., Zary, N., 2015. Design of Mobile Augmented Reality in Health Care Education: A Theory-Driven Framework. *JMIR Medical Education* 1, e10. <https://doi.org/10.2196/mededu.4443>

APÊNDICE 1 – FRAMES DO VÍDEO DE DEMONSTRAÇÃO

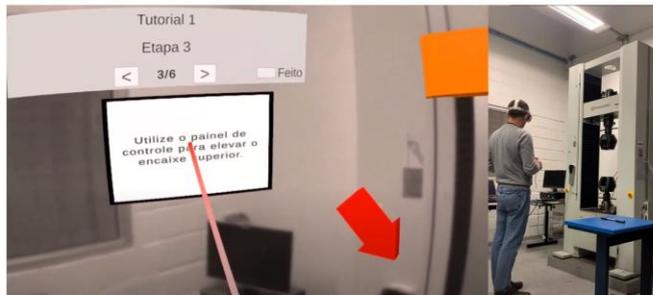
Etapa 1



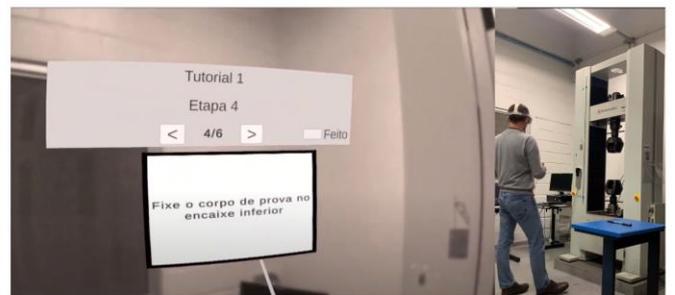
Etapa 2



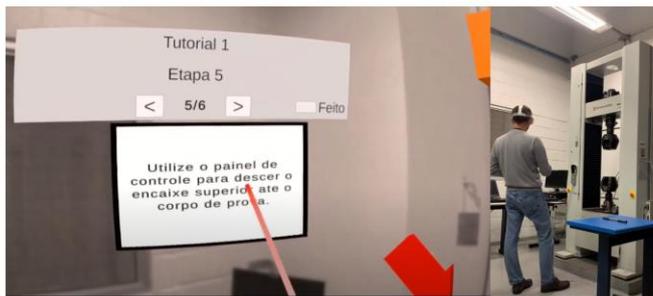
Etapa 3



Etapa 4



Etapa 5



Etapa 6

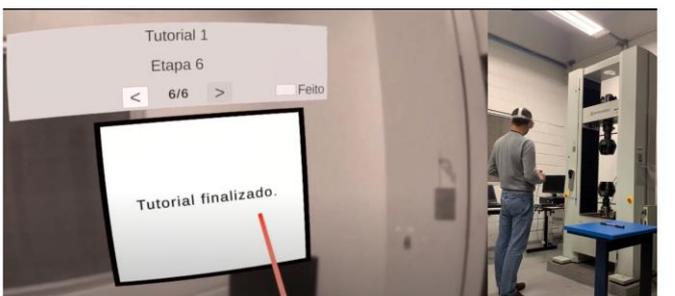


FIGURA 16 - EXIBIÇÃO DOS ELEMENTOS DE CADA ETAPA DO TREINAMENTO DO ESTUDO DE CASO
Fonte: O Autor