

DESENVOLVIMENTO DE BIBLIOTECA PARA PROJEÇÃO EM PIRÂMIDE HOLOGRÁFICA

Lucas Matheus Westphal, Dalton Solano dos Reis – Orientador

Curso de Bacharel em Ciência da Computação

Departamento de Sistemas e Computação

Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brasil

lmwestphal@furb.br, dalton@furb.br

este artigo

Resumo: O seguinte artigo apresenta o desenvolvimento de uma biblioteca para auxílio na construção de cenas utilizando uma pirâmide holográfica. A construção da biblioteca se dá em duas partes: a biblioteca de projeção, e uma aplicação de testes, ambas desenvolvidas em Unity. Foram realizados testes para averiguar o potencial de visibilidade da pirâmide, e para verificar a melhor aproximação para a biblioteca de projeção. A biblioteca se mostrou capaz de gerar uma cena holográfica sem gasto de muitos recursos, além da facilidade de utilização, pois não se tem a necessidade de clonar o objeto que será renderizado, e posicioná-lo manualmente na cena.

Palavras-chave: Ciência da computação. Sistemas de informação. Holografia. Pirâmide holográfica. Unity

1 INTRODUÇÃO

A holografia, desenvolvida por Dennis Gabor em 1947, apesar de ser uma técnica ainda ligeiramente subdesenvolvida segundo Ferreira e Lopes (2017), tem várias aplicações no mundo, tanto científicas quanto artísticas ou em questões de segurança. Holografia (do grego *holos*: todo, inteiro e *graphos*: sinal, escrita), segundo Gabor (1971), é um meio de registro “integral” da informação, com relevo e profundidade, que se obtém a partir da divisão das ondas luminosas, e seu padrão de inferência sobre um objeto, e por esse motivo, um holograma possui propriedades diferentes dos meios tradicionais de visualização da informação.

Segundo Rebordão (1989) os hologramas são popularmente conhecidos como “fotografias tridimensionais (ou 3D)”. Os filmes de ficção científica, as exposições em que têm circulado, vulgarizaram o holograma embora pouco tenham contribuído para elucidar a sua estrutura, construção e aplicações. Se falando na holografia em filmes, é indispensável citar referências como Matrix, Homem de Ferro e Star Wars. Este último já trazia, na década de 70, efeitos especiais de hologramas comunicáveis. Por se tratar de um filme futurista, pode-se perceber como os produtores viam a holografia muito antes de ela ser aperfeiçoada com as tecnologias atuais.

Para Hoffman (2018), é muito importante para a sociedade inovar o método de representação das aplicações do cotidiano, pois o mercado audiovisual está em constante desenvolvimento. O holograma é umas dessas representações, pois ele torna mais imersiva a experiência de observar objetos em 3D. Para tal feito, algumas técnicas de projeção holográfica são utilizadas, sendo as mais conhecidas o Head-Up Display (HUD), o fantasma de Pepper, e a pirâmide holográfica. Esta última técnica, segundo Schivani *et al.* (2018) se trata de um display montado em formato piramidal, que quando posicionado sobre uma superfície luminosa, como um smartphone, tablet ou monitor, dá a impressão de que o objeto observado é tridimensional.

As várias *engines* gráficas atuais, como a Unity, possuem inúmeras rotinas gráficas, desde a representação bidimensional, ou 2D, até representação estereoscópica, porém não se tem uma rotina que auxilie o desenvolvimento de cenas para a projeção holográfica (UNITY, 2019). Uma forma de disponibilizar este auxílio é por meio de bibliotecas, que podem ser em formato de *asset* ou *packages*. Tendo em vista este cenário, este trabalho se propôs a desenvolver uma biblioteca que permite gerar cenas 3D para a projeção holográfica, bem como disponibilizar uma aplicação de testes que inclui outras formas de visualização do modelo, e um guia para montagem da pirâmide.

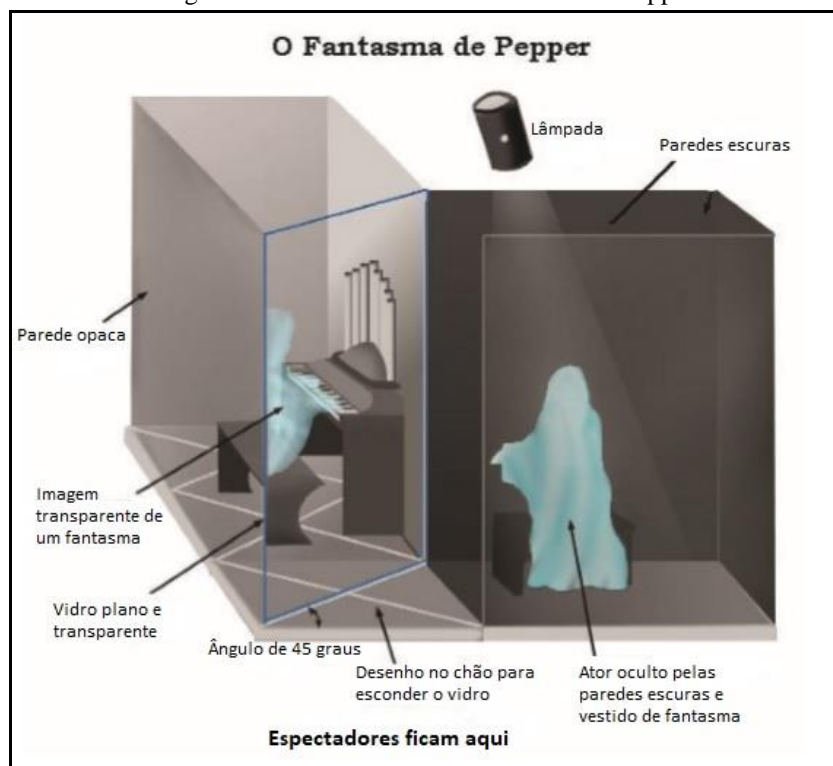
2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção são apresentados conceitos e ferramentas utilizados no desenvolvimento da biblioteca. Serão abordados os seguintes assuntos: fantasma de Pepper, pirâmide holográfica e modelagem 3D. Por fim, são apresentados os trabalhos correlatos a este.

2.1 FANTASMA DE PEPPER

O Fantasma de Pepper, ou “Casa de Monga”, como é conhecido no Brasil, é segundo Medeiros (2006) uma das ilusões de ótica mais fascinantes e conhecidas em todo o mundo. Trata-se da reflexão de um objeto, oculto dos observadores, em uma lâmina de vidro plana inclinada a 45°, desenvolvida pelo professor de Química inglês John Henry Pepper. A Figura 1 demonstra o funcionamento da técnica de Pepper, utilizando como exemplo uma figura fantasmagórica tocando um piano.

Figura 1 - Funcionamento do Fantasma de Pepper



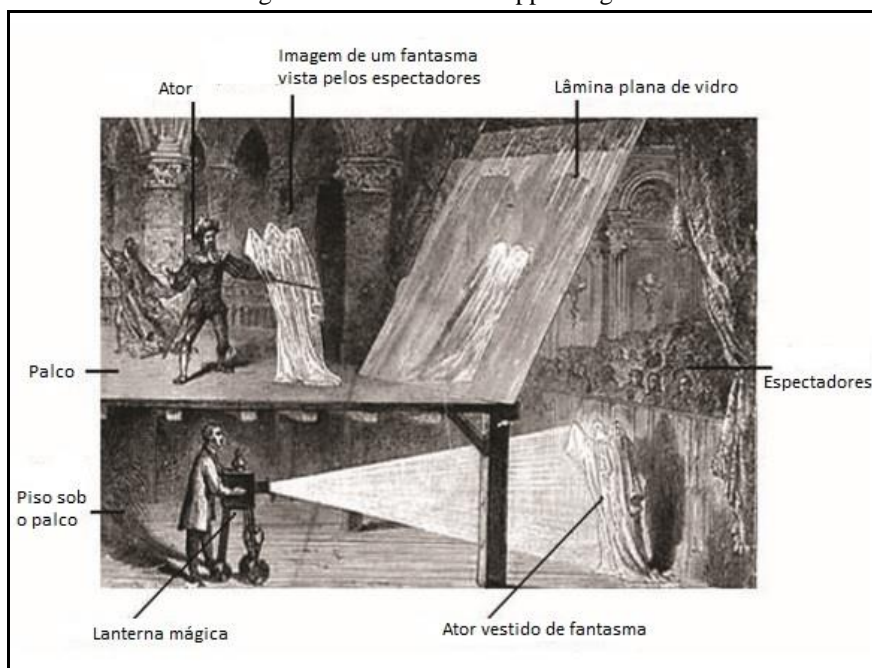
Fonte: Medeiros (2006), alterado pelo autor.

Percebe-se, na Figura 1, que conforme explica Medeiros (2006) existem dois compartimentos de mesmo tamanho, perpendiculares entre si. No primeiro, com as paredes opacas e visível ao público, se encontra apenas um piano, enquanto no compartimento vizinho, com paredes escuras e oculto do público, se encontra um ator vestido de fantasma, encenando tocar o piano. Segundo o autor, a posição do banco a frente do piano e a do ator são equidistantes em relação a lâmina de vidro inclinada a 45° que é mostrada na figura. O autor também aponta que existe algum tipo de obstáculo que não permite que o público se aproxime da cena o suficiente para ver o interior do compartimento onde se encontra o ator.

Inicialmente, segundo Medeiros (2006), o compartimento com o piano é intensamente iluminado, enquanto o outro é deixado as escuras. À proporção que a luz do compartimento com o piano é reduzida, se aumenta a iluminação do compartimento com o ator, fazendo assim com que sua imagem seja gradativamente refletida no vidro inclinado na direção do público, cada vez com mais intensidade. O autor constata também que conforme a iluminação sobre o ator aumenta, mais nítido se torna o reflexo, perdendo deste modo o seu aspecto fantasmagórico. Assim, segundo o autor, é possível produzir imagens com aspectos mais ou menos fantasmagóricos através do controle da iluminação dos ambientes.

Na versão original levada aos palcos no século XIX pelo professor Pepper, segundo Medeiros (2006), a imagem de um fantasma era produzida com o auxílio de uma lanterna mágica, um antigo projetor, posicionada no piso inferior do palco, conforme demonstra a Figura 2. O autor afirma que de acordo com o ângulo de visão dos espectadores, a ilusão produzida pela encenação de Pepper era perfeita, sendo uma das muitas ilusões de ótica que fez sucesso no século XIX e que ainda surpreende quem a observa sem ter conhecimento em ótica.

Figura 2 - Fantasma de Pepper original



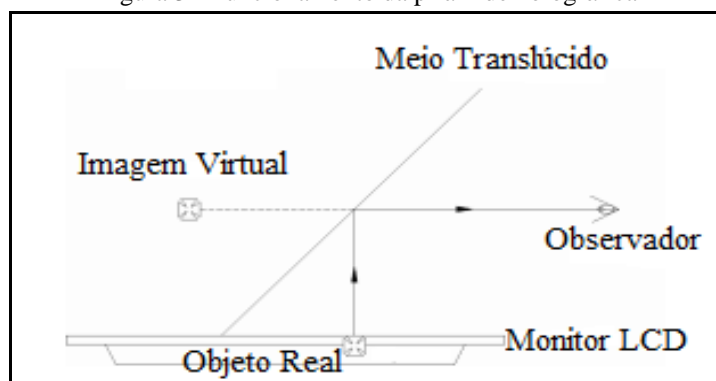
Fonte: Medeiros (2006), alterado pelo autor.

Medeiros (2006) conclui que existe no uso pedagógico da técnica, uma possibilidade inegável de cativar a atenção dos estudantes para a aprendizagem, além da curiosidade histórica dos artefatos mencionados em seu trabalho.

2.2 PIRÂMIDE HOLOGRÁFICA

A pirâmide holográfica é um utensílio, montado geralmente de acrílico, que é utilizado para projetar objetos em 3 dimensões. Segundo Schivani *et al.* (2018), as imagens observadas através das pirâmides utilizam a técnica “Fantasma de Pepper”. A Figura 3 demonstra o funcionamento da pirâmide com a imagem original exibida por um display, que pode ser um smartphone, TV, monitor ou tablet, por exemplo. E que por sua vez é refletida pelas faces da pirâmide, segundo conceitos de Pepper.

Figura 3 - Funcionamento da pirâmide holográfica



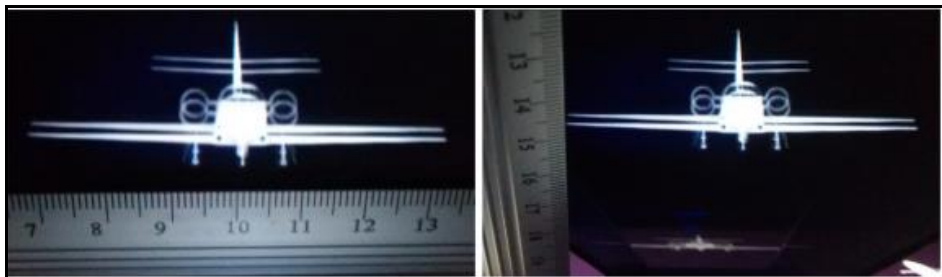
Fonte: Schivani *et al.* (2018).

Para o funcionamento correto da pirâmide, Schiviani *et al.* (2018) destacam que a face central do display deve ser posicionada no topo da pirâmide, e o conjunto pirâmide-display pode ser disposto com a base da pirâmide voltada tanto para baixo como para cima. Cada segmento da pirâmide reflete uma imagem diferente, ou o mesmo objeto com pontos de vista diferentes, simulando um objeto tridimensional.

Conforme constatado por Schivani *et al.* (2018), quando a pirâmide foi feita em proporção para um display pequeno, como um smartphone, não havia problema na projeção da imagem, porém quando as proporções aumentaram, para um monitor de 14 polegadas, ocorreu um efeito de duplicata da imagem, como pode ser visto na Figura 4. Os autores, após realizarem cálculos sobre a refração e reflexão dos raios incidentes sobre as faces da pirâmide chegaram a conclusão de que este efeito é decorrente de um desvio do raio incidente ao sofrer reflexão interna e refração, passando do ar para a pirâmide e de volta para o ar. E devido a espessura do material utilizado, no caso acrílico de 4mm, o desvio

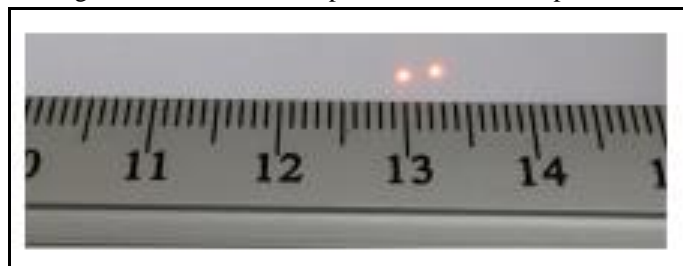
recorrente deste processo se torna mais acentuado, e perceptível a olho nu. A Figura 5 demonstra o efeito de desvio sofrido por um raio laser, respeitando o mesmo ambiente da iluminação original do monitor sobre a pirâmide.

Figura 4 – Efeito de duplicata na pirâmide



Fonte: Schivani *et al.* (2018).

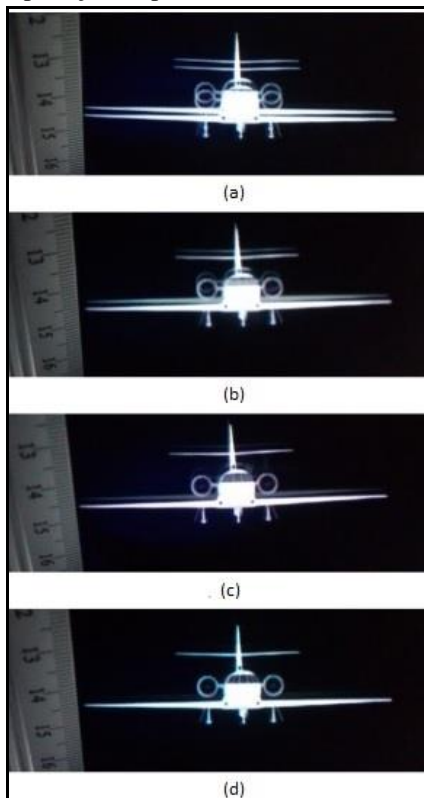
Figura 5 - Desvio sofrido pelos raios de luz na pirâmide



Fonte: Schivani *et al.* (2018).

Para solucionar o problema de duplicata na pirâmide, foram utilizadas por Schivani *et al.* (2018) películas reflexivas, como as presentes em janelas de carros e portas de vidro, por terem a capacidade de diminuir a iluminância das imagens da face oposta da pirâmide, reduzindo o efeito de duplicata. Os autores utilizaram uma película automotiva genérica encontrada em comércio e classificada como #2. Após aplicação dessas películas, notou-se que quanto mais películas eram sobrepostas, mais nítida a imagem parecia, e menos efeito de duplicata era visível. Conforme demonstra a Figura 6, a imagem A representa a pirâmide sem aplicação de película reflexiva, a imagem B representa a pirâmide com aplicação de uma camada de película, a imagem C com duas camadas e a imagem D com três camadas.

Figura 6 - Aplicação de película automotiva sobre a pirâmide



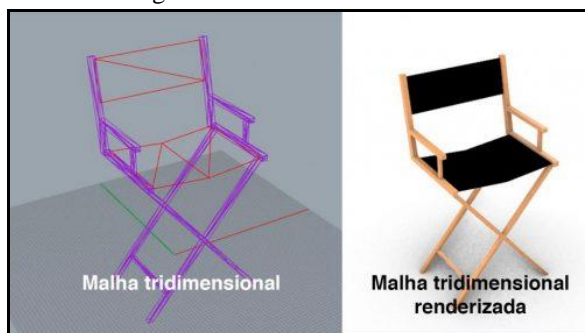
Fonte: Schivani *et al.* (2018), alterado pelo autor.



2.3 MODELAGEM 3D

Modelagem digital tridimensional (modelagem 3D) segundo Rodrigues (2018) é a construção de um desenho ou modelo de três dimensões através de um software 3D, que utiliza modelos matemáticos de representação. O resultado desta construção chama-se malha tridimensional. A Figura 7 demonstra uma malha tridimensional e um modelo renderizado da mesma malha com todas as texturas.

Figura 7 - Malha tridimensional



Fonte: Rodrigues (2018).

Quando se fala em 3D, grande parte das pessoas associa a filmes e desenhos, que, conforme afirma Rodrigues (2018), estão muito realistas nos últimos tempos, entretanto a modelagem 3D não é utilizada apenas para este propósito. A gama de possibilidades de utilização da modelagem 3D é muito ampla, e pode ser utilizada de forma profissional ou não, em diversos segmentos, tais como arquitetura, design, engenharia, jogos, entre outros.

Atualmente, segundo Rodrigues (2018), existe uma grande quantidade de softwares para modelagem, sendo os mais utilizados o 3DS Max, Blender, Sketchup, Maya, Cinema 4D, Zbrush, Rhinoceros. E como motor gráfico e de desenvolvimento de jogos, além de ser um software de modelagem 3D, o Unity. Cada software tem suas características específicas, bem como seus problemas, e a utilização do software depende da área de utilização da modelagem.

2.4 TRABALHOS CORRELATOS

Foram selecionados um produto e dois trabalhos que utilizam da mesma forma de representação gráfica ou *display*. No Quadro 1 é apresentado um produto que simula a captura de criaturas digitais (PIXELSAV, 2015). O primeiro trabalho se trata de um estudo sobre como a holografia pode ser utilizada ao se estudar fenômenos astronômicos (HOFFMAN, 2017), apresentado no Quadro 2. No Quadro 3 é descrito um trabalho sobre a aplicação de holografia utilizada para gerar um avatar da assistente virtual da Microsoft, a Cortana (ARCHER, 2017).

Quadro 1 – Totem holográfico: Pokemon

Referência	Pixelsav (2015)
Objetivos	Simular a captura de criaturas digitais.
Principais funcionalidades	Permitir que o usuário possa utilizar as próprias mãos para interagir com o produto.
Ferramentas de desenvolvimento	Por ser um produto comercial, não foram encontradas especificações técnicas.
Resultados e conclusões	Por ser um produto comercial, não foram encontrados dados de estudo sobre o assunto.

Fonte: elaborado pelo autor.

O Totem holográfico (PIXELSAV, 2015) é um display volumétrico que se utiliza da pirâmide holográfica, dando a impressão de que o objeto, ou criatura, está dentro do recipiente. As criaturas, no caso, são monstros virtuais da animação/jogo Pokémon, pertencente a empresa Nintendo. O usuário interage com o produto, utilizando as próprias mãos para alterar a cena projetada, conceito que a empresa caracteriza como “holografia tátil”. A Figura 8-a demonstra o produto em utilização por um usuário, que realiza a captura do Pokemon tipo fantasma Haunter. A captura do Pokemon se dá arremessando no mesmo uma esfera, chamada Pokebola, que prende a criatura em seu interior (Figura 8-b).

Figura 8 - Utilização do Totem Holográfico



Fonte: Pixelsav (2015).

Por se tratar de um produto comercial, não foram encontrados dados sobre seu desenvolvimento, ou testes e resultados de estudos.

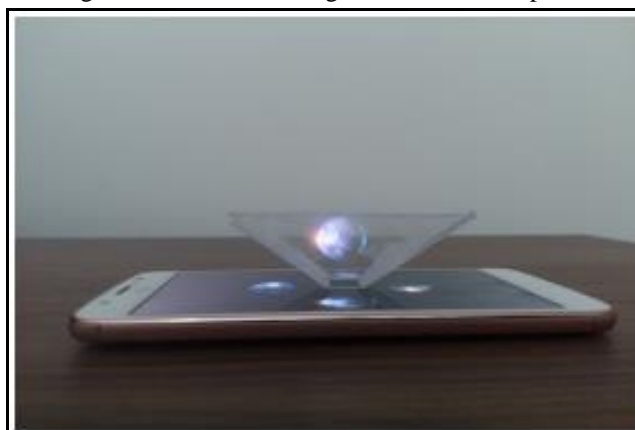
Quadro 2 – Estudo sobre a holografia aplicada a visualização do eclipse solar e lunar

Referência	Hoffman (2017)
Objetivos	Facilitar a visualização dos fenômenos astronômicos eclipse lunar e solar.
Principais funcionalidades	Permitir ao usuário visualizar como ocorre um eclipse em quatro pontos de vista, facilitando o entendimento da posição dos astros durante o fenômeno.
Ferramentas de desenvolvimento	Animação do fenômeno astronômico gerada pelo autor utilizando 3DS Max 2017, software disponibilizado pela Autodesk.
Resultados e conclusões	Os resultados do estudo foram satisfatórios, pois permitiram a visualização em escala reduzida de como ocorre um eclipse, e proporcionou a realização de um projetor holográfico a partir de uma pirâmide construída a partir de utensílios básicos.

Fonte: elaborado pelo autor.

Hoffman (2018) desenvolveu uma aplicação de projeção holográfica para facilitar a visualização do fenômeno astronômico eclipse, tanto lunar quanto solar. Para atingir tal objetivo, Hoffman (2018) utilizou a pirâmide holográfica, e a animação do fenômeno foi gerada pelo próprio autor. A Figura 9 demonstra um protótipo da pirâmide, utilizado como primeira experiência realizada por Hoffman (2018). Para esta experiência, foi utilizado um arquivo de vídeo pronto para a pirâmide, disponível na internet, e a estrutura foi montada em acetato, fixa com fita adesiva para fácil desmontagem caso necessitasse de ajustes estruturais.

Figura 9 – Pirâmide holográfica sobre smartphone



Fonte: Hoffman (2018).

O resultado desta primeira experiência realizada por Hoffman (2018) foi satisfatório, tendo em vista que proporcionou a realização de um projetor holográfico a partir de uma pirâmide construída com utensílios básicos. A animação do fenômeno astronômico foi gerada pelo autor, utilizando o software de modelagem 3DS Max 2017, disponibilizado pela Autodesk. Um protótipo final foi construído pelo autor, utilizando acrílico transparente com espessura de 4mm. A Figura 10 demonstra o protótipo final em utilização.

Figura 10 - Protótipo final da pirâmide de acrílico sobre monitor de LED



Fonte: Hoffman (2018).

Com a utilização do protótipo final, o autor constata o potencial didático proporcionado pelas pirâmides holográficas, gerando uma experiência de aprendizado diferenciada, podendo fomentar maior interesse do público escolar.

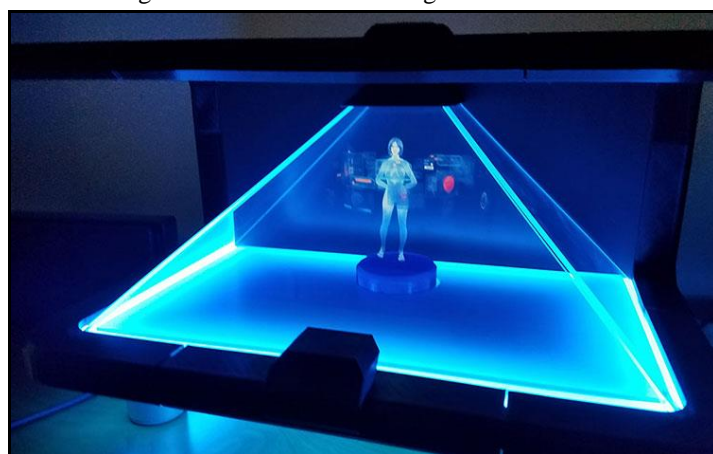
Quadro 3 – *Holographic Cortana appliance: Working concept*

Referência	Archer (2017)
Objetivos	Demonstrar um avatar para a assistente virtual Cortana.
Principais funcionalidades	Permitir que o usuário possa utilizar comandos de voz para interagir com a aplicação. Permitir que o usuário visualize o avatar a partir da pirâmide holográfica.
Ferramentas de desenvolvimento	Aplicação para gerar o avatar feita em Unity. Um serviço <i>proxy</i> que verifica os retornos da Cortana. Iluminação controlada por um microcontrolador Arduino. Aplicação de reconhecimento facial.
Resultados e conclusões	Os resultados foram satisfatórios, apesar dos problemas encontrados. Foi possível fazer pesquisas de base de conhecimento com a Cortana, além poder ser estendido para automação residencial e tarefas de música, por utilizar a Cortana nativa.

Fonte: elaborado pelo autor.

Archer (2017) desenvolveu em seu projeto uma aplicação para apresentar uma versão holográfica da assistente virtual da Microsoft, a Cortana. Segundo o autor, este conceito é o que ele imagina da Cortana ou o Google Home se estes utilizassem a coadjuvante holográfica da franquia de videogame Halo. O projeto utiliza um dispositivo Windows 10 nativo, com 4Gb de memória, e uma placa Arduino para controle das luzes da plataforma. Um monitor de LED USB projeta a animação para a pirâmide. A Figura 11 mostra o projeto construído.

Figura 11 - Conceito de holograma da Cortana



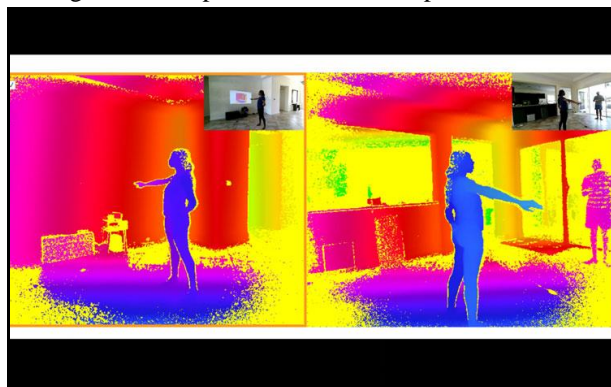
Fonte: Archer (2017).

O software responsável pelo controle da plataforma roda em duas partes. Uma delas é um aplicativo 3D desenvolvido em Unity que apresenta e anima a Cortana em três ângulos de câmera diferentes, e este por sua vez se comunica com o segundo aplicativo, um serviço proxy que analisa os dados que são enviados e que são recebidos pela Cortana, e este renderiza o retorno HTML da Cortana, que então é apresentado pelo aplicativo Unity. Também é

empregada uma câmera, que por meio de reconhecimento facial move a perspectiva de visão da câmera do aplicativo Unity, procurando deixar a aplicação mais tridimensional.

Os movimentos de animação da Cortana foram capturados com o auxílio da esposa de Archer (2017), que se submeteu a várias tomadas de captura de movimento na sala de estar da casa do autor. Para a captura dos movimentos, foram utilizados dois dispositivos Kinect, e a captura foi então inserida na animação da Cortana. A Figura 12 demonstra algumas cenas capturadas.

Figura 12 - Captura de movimento para a Cortana



Fonte: Archer (2017).

Segundo o autor, saber quando a Cortana nativa estava sendo exibida era um grande desafio, por se tratar de um aplicativo da UWP, ou Universal Windows Platform. Isso significa que seu design *sandbox* impede que sejam feitas chamadas de *handle* de janela, para verificar se um aplicativo está iniciado. O autor então solucionou o problema monitorando um único pixel da barra de tarefas do Windows para determinar quando a UI da Cortana estava presente. Por se tratar da Cortana nativa, este dispositivo pode ser usado não apenas para pesquisas de base de conhecimento, mas também para automação residencial e tarefas de música.

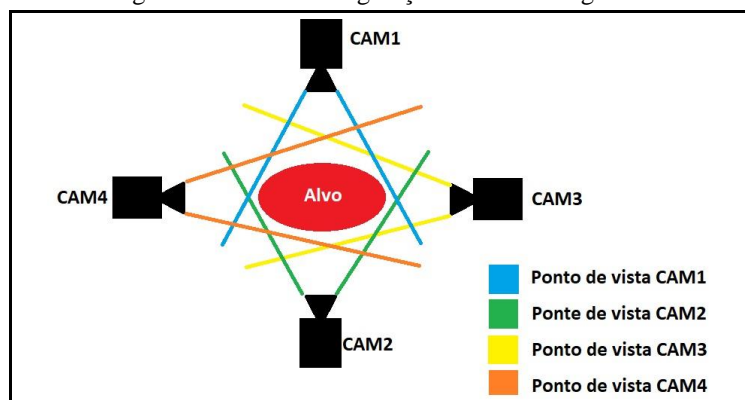
3 DESCRIÇÃO DA BIBLIOTECA

Esta seção tem por objetivo apresentar uma visão geral da funcionalidade da biblioteca desenvolvida, bem como aspectos importantes do seu desenvolvimento. Esta seção está separada em visão geral, montagem da pirâmide holográfica, desenvolvimento da biblioteca e desenvolvimento da aplicação de teste.

3.1 VISÃO GERAL DA BIBLIOTECA

Esta biblioteca tem como principal objetivo facilitar a modelagem de objetos em cenas holográficas, utilizadas em conjunto com a pirâmide holográfica. Ela disponibiliza ao usuário da Unity, em formato de *asset*, os 4 pontos de visão necessários para ser gerada a cena holográfica, como demonstra a Figura 13, sendo possível parametrizar um objeto alvo para a câmera seguir, bem como a distância entre este alvo e as câmeras e se a biblioteca está em formato de configuração.

Figura 13 - Modelo de geração de cena holográfica

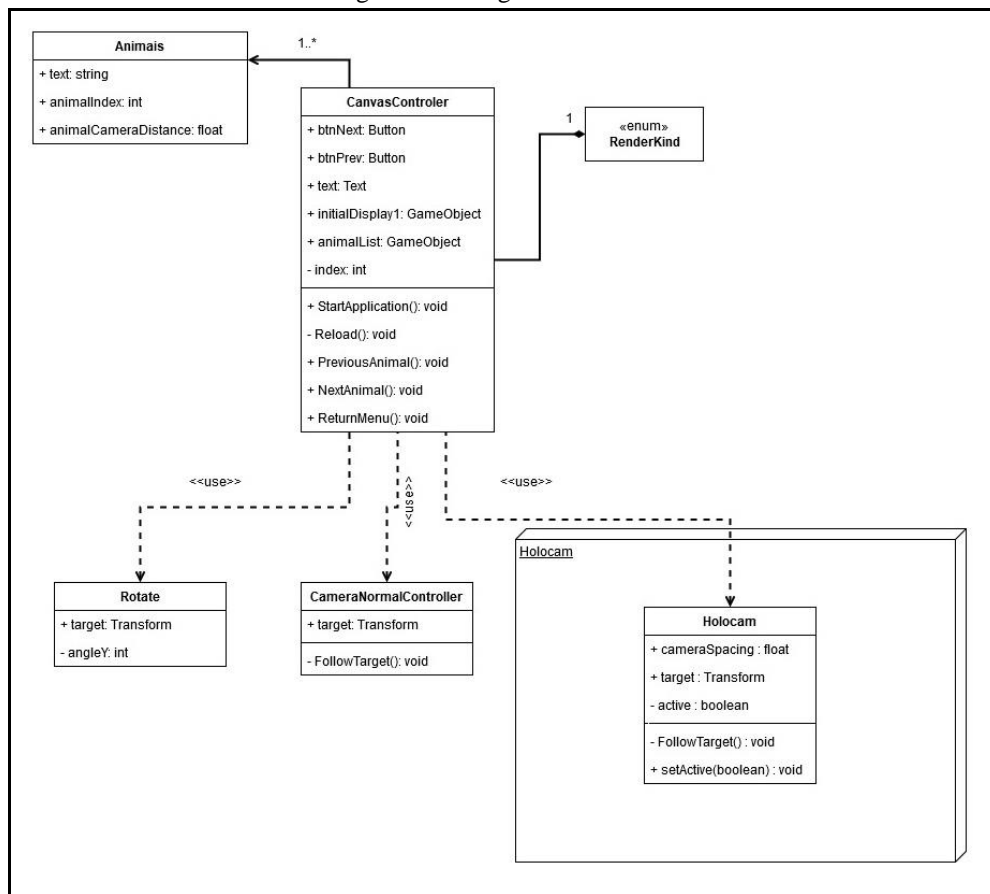


Fonte: Elaborado pelo autor.

Foi desenvolvida em conjunto com a biblioteca uma aplicação de testes que consiste em um compilado de modos de projeção de uma série de modelos 3D de animais, sendo eles o modo normal, projeção na pirâmide holográfica e realidade aumentada (XR). Em cada modo de visualização existem dois botões para seleção do animal em exibição, que vão selecionar o próximo animal na lista e o anterior, um botão para retornar ao menu principal e um texto

informativo sobre o animal selecionado. A Figura 14 exibe o diagrama de classes dessa aplicação, utilizando também a biblioteca desenvolvida.

Figura 14 - Diagrama de classes



Fonte: Elaborado pelo autor.

Como a biblioteca e a aplicação de testes foram desenvolvidas em Unity, entende-se que os métodos `Update()` e `Start()` existem em todas as classes do diagrama, mas foram ocultados. Salvo a classe `Animais`, pois se trata de um `ScriptableObject`, classe utilizada para criar objetos com valores padrão dentro da Unity. Nesse caso, são informados na classe `Animais` os valores do índice do animal dentro da lista de animais que existe em todas as cenas como um `GameObject` com os modelos adicionados dentro dele, o texto informativo sobre o animal e o espaçamento entre o modelo alvo e a câmera da cena.

3.2 MONTAGEM DA PIRÂMIDE

Em primeiro momento foi optado por se produzir um protótipo da pirâmide holográfica para a tela de uma *smartphone* utilizando acrílico. O material foi selecionado pois é de fácil acesso, sendo encontrado em capas transparentes de CDs. Foram recortadas 4 faces com as dimensões de 3 centímetros (cm) de topo, 6 cm de base e 5 cm de altura. Optou-se por fixar a pirâmide com fita adesiva para eventuais ajustes na estrutura da pirâmide. A Figura 15 demonstra um teste feito utilizando uma animação do Pokémon Charmander encontrada no YouTube.

Figura 15 - Teste utilizando a pirâmide de acrílico

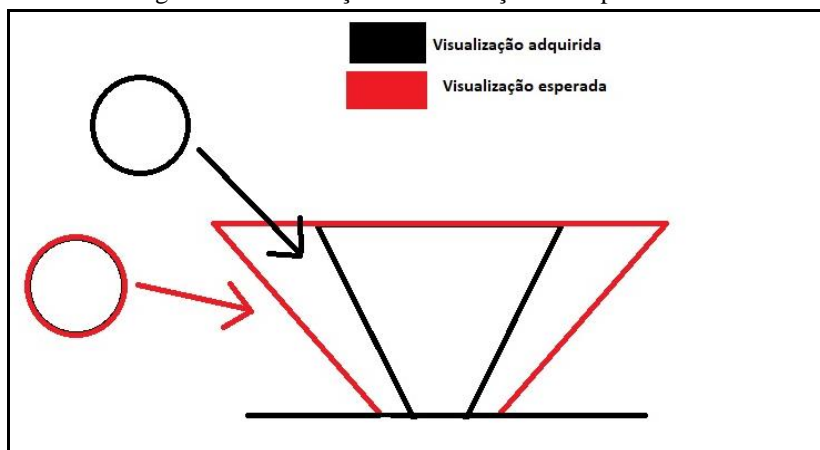


Fonte: Elaborado pelo autor.



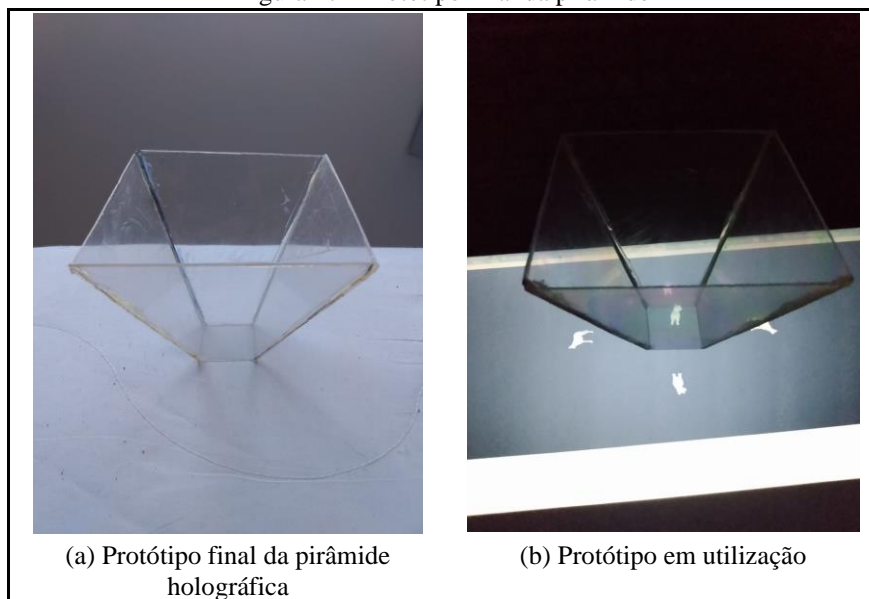
Após a finalização deste teste, passou-se então à construção de uma versão maior da pirâmide. Seguindo os cálculos de Hoffman (2017) para a montagem correta da pirâmide para o tamanho de tela de 15 polegadas, observou-se que o material não iria ser suficiente para as 4 faces, então foi diminuído o tamanho da base da face, consequentemente aumentando o ângulo em relação a tela. Essa alteração faz com que o usuário tenha que olhar para a face da pirâmide mais do alto, como demonstra a Figura 16. O protótipo final da pirâmide pode ser visto sozinho (Figura 17-a) e em utilização (Figura 17-b).

Figura 16 - Diferença de visualização entre pirâmides



Fonte: Elaborado pelo autor.

Figura 17 - Protótipo final da pirâmide



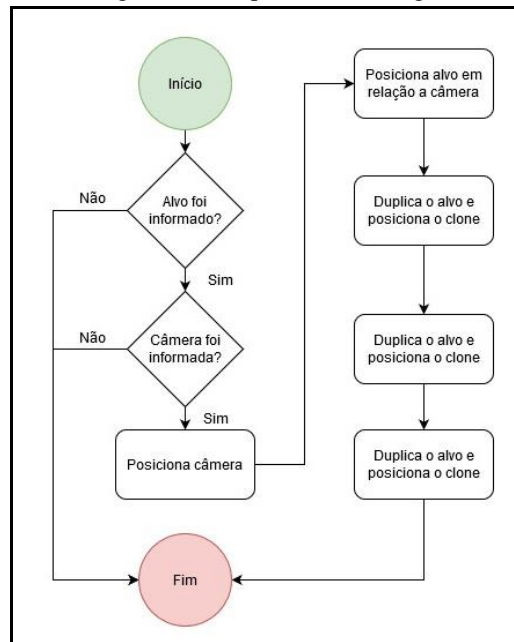
Fonte: Elaborado pelo autor.



3.3 ASSET DE PROJEÇÃO HOLOGRÁFICA

Para o *asset* de projeção holográfica, inicialmente foi optado por um código que seria adicionado a cena em que o objeto que seria projetado se encontra. O código seria responsável por clonar o objeto três vezes, e posicionar os objetos resultantes e a câmera da cena (Figura 18).

Figura 18 - Diagrama de sequência do código de clonagem



Fonte: Elaborado pelo autor.

Para tornar a explicação menos repetitiva, a seguinte legenda foi elaborada (Tabela 1).

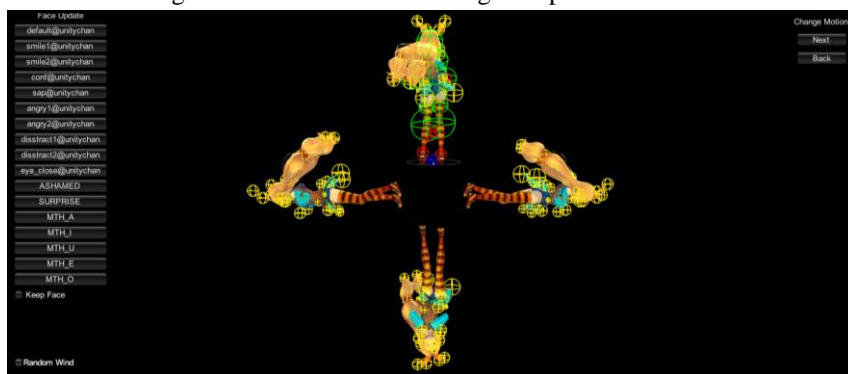
Tabela 1 - Legenda de abreviações

Abreviação	Significado
XCam	Coordenada X da câmera
YCam	Coordenada Y da câmera
ZCam	Coordenada Z da câmera
XAlvo	Coordenada X do alvo original
YAlvo	Coordenada Y do alvo original
ZAlvo	Coordenada Z do alvo original

Fonte: Elaborado pelo autor.

O posicionamento dos objetos se dá da seguinte forma: a câmera é posicionada nas coordenadas $X = 0$, $Y = 0.5$ e $Z = -1$ na cena, e sua rotação é zerada em todos os eixos. Após isto, o alvo é posicionado com $X = X_{Cam}$, $Y = Y_{Cam} + 0.3$ e $Z = Z_{Cam} + 3$, e sua rotação se iguala a da câmera. Feito esse passo, o primeiro clone é instanciado e posicionado em $X = X_{Cam}$, $Y = Y_{Cam} - 0.3$ e $Z = Z_{Alvo}$. O objeto clonado também é rotacionado 180° em X. Para o segundo clone, sua posição fica em $X = X_{Cam} + 0.3$, $Y = Y_{Cam}$ e $Z = Z_{Alvo}$, e ele é rotacionado 90° em X e -90° em Z. O último clone é então instanciado, posicionado em $X = X_{Cam} - 0.3$, $Y = Y_{Cam}$ e $Z = Z_{Alvo}$, e é rotacionado 90° em X e Z. O resultado desta operação pode ser visto na Figura 19.

Figura 19 - Resultado da clonagem e posicionamento



Fonte: Elaborado pelo autor.

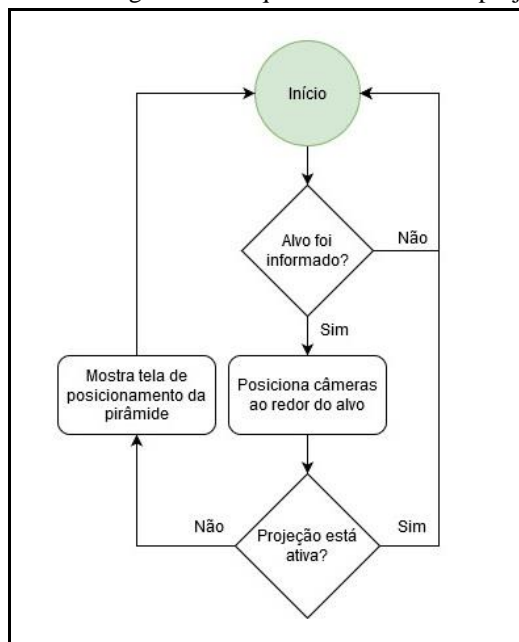
Porém, quando mais testes foram realizados com o código, o resultado não foi satisfatório, pois conforme foram adicionados mais objeto a cena, mais cópias do código eram requeridas, e os objetos eram adicionados todos na

mesma posição, causando uma sobreposição. Foi pensado em uma forma de corrigir o problema utilizando mais variáveis na clonagem, para espaçamento entre objetos, mas isso não iria resolver o fato de que quanto mais objetos para clonar, mais custoso se tornaria o trabalho de gerar a cena holográfica. Também é importante dizer que se o objeto tivesse alguma animação controlada pelo usuário, apenas o objeto alvo original iria executar esta animação.

Diante deste resultado, foi iniciado o desenvolvimento de um segundo *asset*, utilizando uma técnica muito comum no desenvolvimento de minimaps em jogos de *first-player shooter* (FPS) e *role-playing game* (RPG), onde se captura o retorno de uma câmera na cena e direciona este resultado a uma imagem. Desta forma não são instanciados mais objetos, foi criado um *prefab* que consiste em 4 câmeras (uma câmera para cada face da pirâmide), um *canvas* com quatro imagens que irão receber o retorno das câmeras e uma imagem estática de posicionamento da pirâmide. Com esta aproximação pode-se perceber que as animações dos objetos eram refletidas em todos os pontos de vista, e o desempenho da cena holográfica melhorou, pois não era mais necessário gastar recursos clonando os objetos da cena.

Conforme demonstra a Figura 20, a biblioteca inicia verificando se existe um alvo fixo para a biblioteca focar, se este for o caso, as câmeras são posicionadas em torno do alvo, levando em conta o espaçamento entre o alvo e as câmeras informado pelo usuário. Caso contrário elas permanecem na posição original, sendo esta onde o usuário as posicionou no diagrama de cena da Unity. Caso a projeção não esteja habilitada, é demonstrada uma tela de posicionamento da pirâmide.

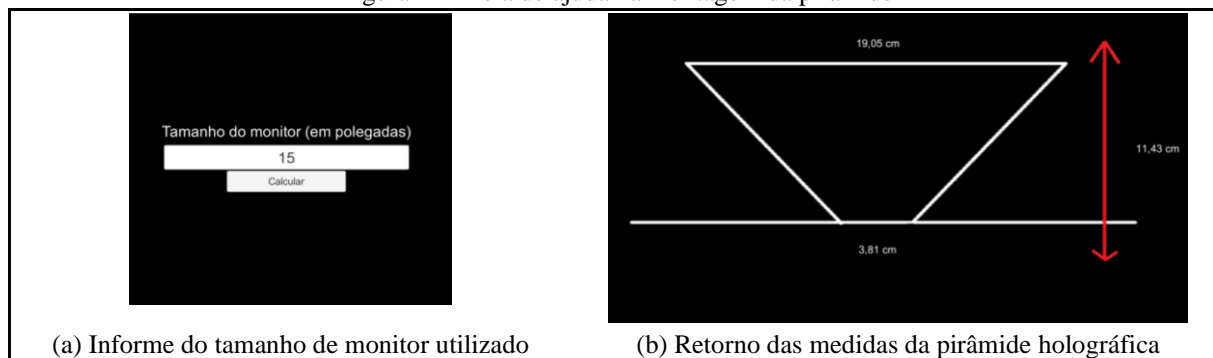
Figura 20 - Diagrama de sequência do modo de projeção



Fonte: Elaborado pelo autor.

Foi desenvolvido em conjunto com o *prefab* uma cena de ajuda para a montagem correta da pirâmide, onde o usuário informa o tamanho do monitor utilizado para a projeção e são feitos os cálculos desenvolvidos por Hoffman (2017) para as medidas do topo, base e altura da pirâmide. A cena pode ser vista em utilização na Figura 21, onde é informado que se é utilizado um monitor de 15 polegadas (Figura 21-a) e são retornados os valores das medidas relativas (Figura 21-b)

Figura 21 - Tela de ajuda na montagem da pirâmide

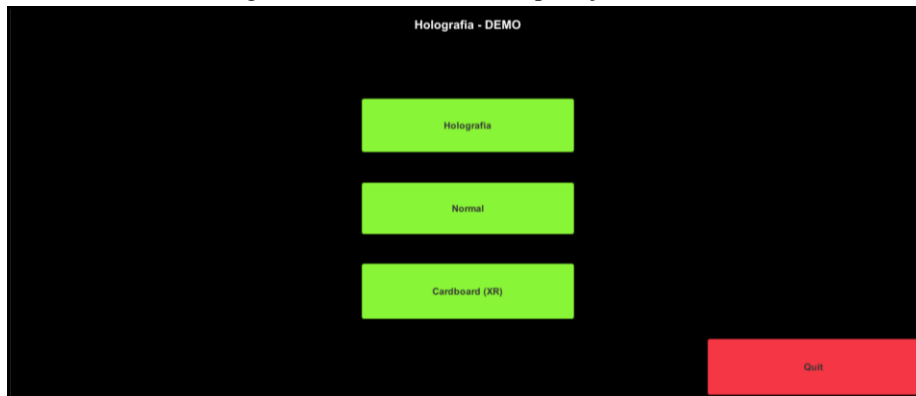


Fonte: Elaborado pelo autor.

3.4 APLICAÇÃO DE TESTE

Para a utilização desta aplicação são necessários dois monitores, um para o controle e um para a projeção holográfica, sendo que quando não se está no modo holografia, o segundo monitor se mantém em um “estado neutro”, onde a tela fica escura e não tem nada na mesma. A Figura 22 ilustra o menu principal da aplicação.

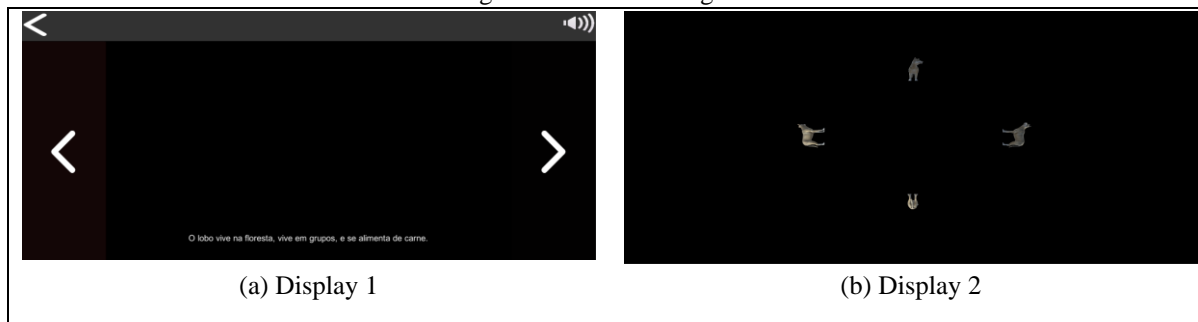
Figura 22 - Menu inicial da aplicação de teste



Fonte: Elaborado pelo autor.

A opção Holografia disponibiliza uma cena holográfica utilizando a biblioteca desenvolvida para gerar a projeção no monitor secundário do animal selecionado (Figura 23-a), e os controles da cena no monitor primário (Figura 23-b). Os controles da cena utilizam o script `CanvasController` para funcionar. O método `NextAnimal()` e `PreviousAnimal()` mudam o animal selecionado, e utilizam o método `Reload()`, que atualiza a tela para mostrar o novo animal selecionado, bem como o texto informativo.

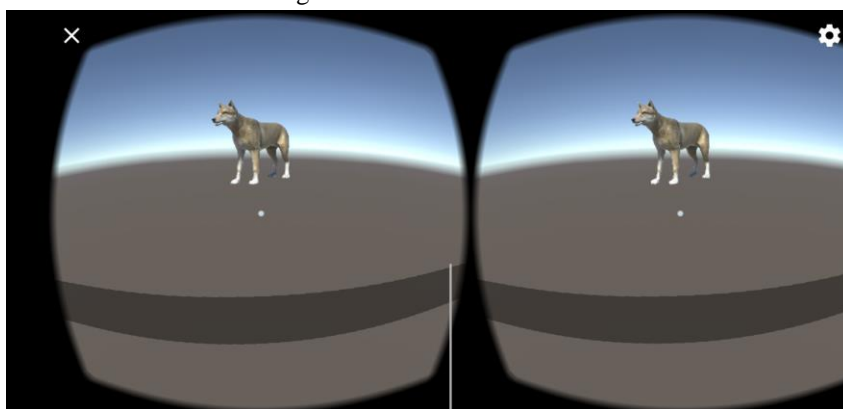
Figura 23 - Modo holografia



Fonte: Elaborado pelo autor.

A opção de Cardboard exige a utilização de um dispositivo móvel compatível com o aplicativo Google Cardboard. Nesse modo de visualização, o usuário utiliza um óculos de realidade aumentada para escolher um animal e visualizá-lo, conforme demonstra a Figura 24. Como a posição da câmera nesse modo de visualização é fixa, optou-se por rotacionar o animal selecionado para o usuário poder visualizá-lo de todos os ângulos. Este comportamento é controlado pelo script `Rotate`. O distanciamento entre a câmera e o modelo do animal é controlado pelo script `NormalCameraController`.

Figura 24 - Modo CardBoard



Fonte: Elaborado pelo autor.

A opção Normal apresenta uma cena simples do modelo do animal selecionado exibido no centro da tela. Nesta opção é utilizado apenas o monitor primário, os controles e o texto informativo estão situados na mesma tela que o animal. Também foi optado por rotacionar o animal nesta cena, pois a câmera é fixa e o distanciamento da câmera também é controlado via script. A Figura 25 demonstra a cena normal em utilização.

Figura 25 - Cena normal da aplicação de teste



Fonte: Elaborado pelo autor.

4 RESULTADOS

A seção de resultados foi dividida em três partes. A primeira trata dos testes com a parte física, ou seja, a pirâmide holográfica. A segunda engloba testes feitos com a biblioteca de projeção. Por fim, são apresentados os testes da aplicação que utiliza a biblioteca de projeção desenvolvida, comparando com os outros modos de visualização (normal e CardBoard).

4.1 TESTES DE HARDWARE



Primeiramente, após a pirâmide estar montada, foi testada a capacidade de se visualizar o objeto na pirâmide em ambientes diferentes. Foram selecionados dois casos para teste: um ambiente bem iluminado, e um ambiente sem iluminação. Para realizar os testes, foi utilizada a imagem do Pokemon Charmander vista na Figura 15. Tendo em vista que a pirâmide holográfica utiliza os mesmos conceitos de reflexão da luz que a técnica de Pepper, o objeto ficou mais nítido e fácil de ver em um ambiente mais escuro, pois a visibilidade do objeto na pirâmide depende da intensidade de brilho da tela em que a pirâmide está posicionada.

Após isso, foram testadas as cores mais visíveis na pirâmide, utilizando uma aplicação em Unity que demonstra um cubo mudando de cor na cena. Esses testes foram inconclusivos, pois o autor vê um espectro de cores melhor diferente do testador, que vê um espectro de cores diferente melhor. Porém em ambos os casos se tem uma predominância de cores do espectro quente, como vermelho, amarelo e laranja, pois se trata de cores mais claras.

4.2 TESTES DE SOFTWARE

Em primeiro momento foi testada a capacidade de clonar objetos na cena com o script de clonagem. Quando se tem apenas um objeto estático na cena, não ocorrem dificuldades em gerar a cena. Porém, se o objeto tiver uma animação controlada pelo usuário, apenas o objeto original irá reproduzir a animação, sendo que os clones se mantêm estáticos. Outra dificuldade encontrada pelo script de clonagem foi que quanto mais objetos para serem clonados, mais custoso se torna a operação, tendo em vista que são gerados 3 novos objetos de cada alvo, e são todos posicionados no mesmo local, causando uma sobreposição.

Com o *prefab* de projeção holográfica se tem um tempo de renderização relativamente baixo, sendo que não é mais necessário clonar os objetos. Porém o alvo da projeção se torna único ou nenhum. A possibilidade também de se ver o cenário em volta do objeto alvo da projeção.



4.3 TESTES FINAIS

Em relação aos modos de visualização normal e Cardboard, vê-se que a projeção em pirâmide holográfica é significativamente mais interativa, pois o usuário tem a possibilidade de andar em volta do projetor, e como foi demonstrado é possível deixar o controle da cena em outra tela, não interferindo na projeção holográfica.

5 CONCLUSÕES

Diante dos resultados apresentados conclui-se que, de modo geral, a biblioteca cumpre sua proposta, pois é possível gerar uma cena com os pontos de vista necessários para a pirâmide holográfica, sem a necessidade de copiar o objeto, colar e posicioná-lo na cena manualmente. Sem esse fator de dificuldade, a biblioteca auxilia no desenvolvimento de cenas para projeção holográfica.

Em comparação com os trabalhos correlatos, exceto o totem holográfico comercializado pela PixelSav (2015) que não foram encontrados dados técnicos, vê-se uma facilidade na criação de cenas holográficas, pois não se faz necessário a clonagem e posicionamento manual dos objetos, ou da utilização da técnica de minimapa pura, pois o *prefab* desenvolvido faz automaticamente esse processo.

Para estender o desenvolvimento da biblioteca, é possível deixar parametrizável quais faces da pirâmide podem ser visualizados e quais devem ser ocultos, bem como deixar visível o cenário por trás do objeto alvo da projeção.

REFERÊNCIAS

ARCHER, J. **Holographic Cortana Appliance: Working Concept**. [S. l.], [2017]. Disponível em <http://untitled.com/blog/cortana-hologram-working-concept>. Acesso em 27 out. 2019.

FERREIRA, C. LOPES, D. **Holografia**. 2017. 8 f. Artigo – Instituto Superior Técnico, Lisboa.

GABOR, D. **Holography**. 1971. 35 f. Palestra (Nobel de Física) - Imperial College of Science and Technology, Londres.

HOFFMAN, B. V. **Um estudo sobre a holografia aplicada a visualização do eclipse solar e lunar**. 2018. 61 f. Trabalho de Conclusão de Curso (Bacharel em Tecnologias Digitais) – Universidade de Caxias do Sul, Caxias do Sul.

MEDEIROS, A. A história e a física do Fantasma de Pepper. **Caderno brasileiro de ensino de física**, UFSC, v. 23, n. 3, p. 329-345, dez, 2006.

PIXELSAV. **Holografia: Pokemon**. Curitiba, 2015. Disponível em: http://www.pixelsav.com.br/projetos/pokemon_pixel/. Acesso em: 20 ago. 2019.

REBORDÃO, J. M. Holografia: Física e Aplicações, **Colóquio/Ciências Revista de Cultura Científica**, Lisboa, n. 4, p.18-34, 1989.

RODRIGUES, A. **O que é modelagem digital ou modelagem 3D**. [S. l.], 2018. Disponível em <http://mundodesenhodigital.com.br/o-que-e-modelagem-digital-ou-modelagem-3d-e/>. Acesso em 02 nov 2019.

SCHIVANI, M. SOUZA, G. F. PEREIRA, E. Pirâmide “holográfica”: erros conceituais e potencial didático, **Revista Brasileira de Ensino de Física**, [S. l.], v. 80, n. 2, nov, 2018.

UNITY, **Unity User Manual (2019.2)**, [S. l.], [2019]. Disponível em <https://docs.unity3d.com/Manual/index.html>. Acesso em 20 out. 2019.