

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO  
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE  
DEPARTAMENTO DE COMPUTAÇÃO**

**FELIPE FERNANDES SECATO**

**APLICAÇÃO DE REALIDADE AUMENTADA SEM  
MARCADORES PARA A VISUALIZAÇÃO DE PRODUTOS  
ONLINE ATRAVÉS DE UMA APLICAÇÃO MÓVEL**

**ALEGRE - ES**

**2017**

FELIPE FERNANDES SECATO

**APLICAÇÃO DE REALIDADE AUMENTADA SEM  
MARCADORES PARA A VISUALIZAÇÃO DE PRODUTOS  
ONLINE ATRAVÉS DE UMA APLICAÇÃO MÓVEL**

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. M.Sc. Jacson Rodrigues Correia da Silva.

ALEGRE - ES

2017

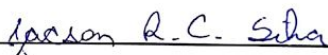
FELIPE FERNANDES SECATO


**APLICAÇÃO DE REALIDADE AUMENTADA SEM MARCADORES  
PARA A VISUALIZAÇÃO DE PRODUTOS ONLINE ATRAVÉS DE  
UMA APLICAÇÃO MÓVEL**


Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Aprovado em 13 de dezembro de 2017.

**COMISSÃO EXAMINADORA**

  
\_\_\_\_\_  
**Prof. M.Sc. Jacson Rodrigues Correia da Silva**  
Universidade Federal do Espírito Santo  
Orientador

  
\_\_\_\_\_  
**Prof. M.Sc. Edmar Hell Kampke**  
Universidade Federal do Espírito Santo

  
\_\_\_\_\_  
**Prof. Dr. Antonio Almeida de Barros Junior**  
Universidade Federal do Espírito Santo

## **AGRADECIMENTOS**

Agradeço primeiramente a minha família, especialmente minha mãe Mariluci Fernandes Secato e ao meu pai Paulo Roberto Secato, pelo carinho e educação que me deram, por sempre estarem ao meu lado me apoiando e me aconselhando nos momentos mais difíceis.

Aos meus amigos do curso, especialmente a galera da CDQ, pelos momentos de risadas, diversão e desabafos. Um agradecimento especial ao meu amigo de república Udson Domingos por sempre segurar a barra e resolver os problemas da casa nos momentos mais complicados.

Ao Paulo, Gilberto, Kleyton, Keila e todos que trabalharam ou frequentavam a seção de tecnologia da informação da UFES, pelo café nosso de cada dia, pela paciência, pela experiência de trabalho adquirida e pelos conselhos que ajudaram no meu processo de amadurecimento.

A Universidade Federal do Espírito Santo, por disponibilizar todos recursos necessários para oferecer um ensino de qualidade, um agradecimento especial a todos os professores do departamento de computação por terem contribuído com minha formação profissional e pessoal, ao meu orientador Jacson Rodrigues Correia da Silva pelos ensinamentos e empenho com o meu trabalho.

A todos que fizeram parte desta etapa da minha vida, muito obrigado!

## RESUMO

A Realidade Aumentada é uma tecnologia que tem ganhado muito espaço nos últimos anos principalmente por conta da popularidade e facilidade de acesso dos dispositivos móveis, mas o uso mais comum da Realidade Aumentada ainda é muito restrito a ações de marketing e jogos. Este trabalho teve como objetivo avaliar o uso da Realidade Aumentada sem marcadores como alternativa na visualização de produtos de uma loja virtual, foi desenvolvida uma aplicação móvel capaz de obter um modelo 3D na Internet e utilizá-lo com a realidade por meio da técnica de rastreamento SLAM, possibilitando assim que o aplicativo funcione em qualquer ambiente sem a necessidade de um conhecimento prévio do local. A validação da aplicação foi feita por meio de questionário qualitativo, onde foi analisado questões como facilidade de uso, principais problemas encontrados e utilidade da aplicação. A Realidade Aumentada sem uso de marcadores se mostrou uma alternativa viável para visualização de produtos online, possibilitando que o usuário visualize o produto previamente em seu próprio ambiente, antes mesmo de realizar a compra.

Palavras-chave: Realidade aumentada sem marcadores, RA, *Simultaneous localization and mapping*, Computação móvel, Comércio eletrônico.

## LISTA DE FIGURAS

Figura 1 - Sensorama.	14
Figura 2 - Sword of Damocles.	15
Figura 3 - Videoplace por Myron Krueger.	15
Figura 4 - Simulador de trem desenvolvido pela USP para a empresa VALE S.A.	16
Figura 5 - Mesa de projeção utilizada na copa do mundo 2014.	17
Figura 6 - Realidade Aumentada em dispositivos móveis.	18
Figura 7 - Realidade-Virtualidade contínua.	19
Figura 8 - Demonstração da Realidade-Virtualidade Contínua.	20
Figura 9 - Aplicações de Realidade Aumentada baseada em monitor.	21
Figura 10: Vantagens e desvantagens entre os sistemas de RA.	22
Figura 11 - Exemplos de marcadores fiduciais.	25
Figura 12 - Taxonomia das técnicas sem marcadores baseadas em modelos.	26
Figura 13 - Comparação das técnicas de rastreamento baseado em modelo.	27
Figura 14 - Bundler: Reconstrução 3D a partir de uma coleção de imagens	28
Figura 15 - Pipeline do SfM.	29
Figura 16 - SLAM: detecção de marcos pontuais (esquerda) e retas (direita).	31
Figura 17 - Arquitetura dos sistemas de RA em dispositivos móveis.	33
Figura 18 - Objeto tridimensional representado por meio de malha de polígonos.	38
Figura 19 - Estruturas de dados de um cubo tridimensional.	39
Figura 20 - Classificação da aplicação desenvolvida.	41
Figura 21 - Protótipo da página principal da loja.	42
Figura 22 - Protótipo da página de visualização de um produto.	43
Figura 23 - Visual da loja virtual.	44
Figura 24 - Visual da loja virtual quando acessado de um dispositivo móvel.	44
Figura 25 - Pacote de modelos 3D utilizado neste trabalho.	46
Figura 26 - Organização dos arquivos da aplicação.	47
Figura 27 - Elementos da aplicação no Unity	48

Figura 28 - Cena DownloadModel visualizada pelo editor do Unity.	50
Figura 29 - Demonstração da cena DownloadModel.	50
Figura 30 - Cena RA visualizada pelo editor do Unity.	51
Figura 31 - Demonstração da cena RA.	52
Figura 32 - Demonstração da cena RA.	53
Figura 33 - Fluxo de uso da aplicação.	54
Figura 34 - Respostas da questão 1 do questionário.	55
Figura 35 - Respostas da questão 2 do questionário.	56
Figura 36 - Respostas da questão 3 do questionário.	57
Figura 37 - Respostas da questão 4 do questionário.	58
Figura 38 - Respostas da questão 5 do questionário.	59
Figura 39 - Respostas da questão 6 do questionário.	59
Figura 40 - Lógica de projeção do modelo 3D pela aplicação.	60
Figura 41 - Respostas da questão 7 do questionário.	62
Figura 42 - Respostas da questão 8 do questionário.	63

## **SUMÁRIO**

<b>1 INTRODUÇÃO</b>	<b>9</b>
<b>1.1 O PROBLEMA E SUA IMPORTÂNCIA</b>	<b>10</b>
<b>1.2 OBJETIVOS</b>	<b>11</b>
<b>1.2.1 Objetivo geral</b>	<b>11</b>
<b>1.2.2 Objetivo específicos</b>	<b>11</b>
<b>1.3 ESTRUTURA DO TRABALHO</b>	<b>11</b>
<b>2 REVISÃO DA LITERATURA</b>	<b>12</b>
<b>2.1 REALIDADE VIRTUAL</b>	<b>12</b>
<b>2.1.1 Breve Histórico</b>	<b>13</b>
<b>2.1.2 Classificação dos sistemas de Realidade Virtual</b>	<b>16</b>
<b>2.2 REALIDADE AUMENTADA</b>	<b>19</b>
<b>2.2.1 Classificação dos sistemas de Realidade Aumentada</b>	<b>20</b>
<b>2.2.2 Classificação das técnicas de interação</b>	<b>22</b>
<b>2.2.3 Técnicas de rastreamento</b>	<b>23</b>
<b>2.2.3.1 Rastreamento com marcadores</b>	<b>24</b>
<b>2.2.3.2 Rastreamento sem marcadores</b>	<b>25</b>
<b>2.3 REALIDADE AUMENTADA EM DISPOSITIVOS MÓVEIS</b>	<b>32</b>
<b>2.3.1 FERRAMENTAS DE RA PARA DISPOSITIVOS MÓVEIS</b>	<b>35</b>
<b>2.3.1.1 ARToolkit</b>	<b>35</b>
<b>2.3.1.2 Vuforia</b>	<b>36</b>
<b>2.3.1.3 Kudan</b>	<b>36</b>
<b>2.3.1.4 Wikitude</b>	<b>37</b>
<b>2.4 MODELAGEM 3D</b>	<b>37</b>
<b>3 METODOLOGIA</b>	<b>41</b>
<b>3.1 DESENVOLVIMENTO DA LOJA VIRTUAL</b>	<b>42</b>
<b>3.2 AQUISIÇÃO DOS MODELOS 3D DOS PRODUTOS</b>	<b>45</b>
<b>3.3 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL</b>	<b>46</b>
<b>3.3.1 Cena DownloadModel</b>	<b>49</b>
<b>3.3.2 Cena RA</b>	<b>50</b>



<b>3.4 DISPONIBILIZAÇÃO DA APLICAÇÃO NA PLAY STORE</b>	<b>53</b>
<b>3.5 UTILIZAÇÃO DO APLICATIVO E AVALIAÇÃO DOS USUÁRIOS</b>	<b>54</b>
<b>4. ANÁLISE DOS RESULTADOS</b>	<b>55</b>
<b>5. CONCLUSÕES</b>	<b>64</b>
<b>REFERÊNCIAS</b>	<b>65</b>
<b>APÊNDICES</b>	<b>70</b>
<b>APÊNDICE I - MÉTODO DE DOWNLOAD DO MODELO 3D</b>	<b>70</b>
<b>APÊNDICE II - PRINCIPAIS FUNÇÕES DO OBJETO RA</b>	<b>71</b>
<b>APÊNDICE III - QUESTIONÁRIO QUALITATIVO</b>	<b>72</b>

## 1 INTRODUÇÃO

Atualmente as compras por lojas virtuais é uma realidade cotidiana na vida de diversas pessoas, a possibilidade de comparar os preços rapidamente, buscar produtos similares e ofertas se torna um grande atrativo para o consumidor. Apesar desse número de compras ter diminuído no Brasil em 2016, estudos mostram que esse valor deve dobrar até 2021 e que esse avanço será possível graças a novos consumidores, isto é, pessoas que até então ainda não estão acostumadas a comprar pela Internet. A maior parte das vendas atualmente das lojas virtuais são de eletrônicos, livros e eletrodomésticos, ocorre pois o público brasileiro está acostumado a comprar produtos que são fáceis de se comparar. A expectativa é que o consumidor aumente sua confiança ao comprar em outras categorias como móveis e roupas (OLIVEIRA, 2016).

Segundo Fachinetto (2013), para atender melhor o público e se manterem competitivas as empresas têm disponibilizado um número maior de produtos e material publicitário (fotos, vídeos) e especificações técnicas de forma a atender mais especificamente cada usuário. Contudo, esses recursos são limitados já que nenhum deles é capaz de permitir uma visualização do produto no ambiente real do cliente. De acordo com Zorzal (2009) a Realidade Aumentada (RA) transporta o ambiente virtual para o espaço de domínio do usuário sendo que a interação entre esses ambientes (físico e virtual) acontece de forma natural, exigindo pouco ou nenhum treinamento.

Devido ao aumento nas vendas dos dispositivos móveis e aumento no poder de processamento desses aparelhos a RA tem se tornado cada vez mais popular, pois técnicas mais recentes, como a *simultaneous localization and mapping* (SLAM), permitem que a RA seja aplicada em cenários não controlados sem o uso de marcadores, prova disso foi o jogo Pokémon Go cujo grande parte do sucesso foi devido ao uso de RA.

Dessa maneira o uso da RA com dispositivos móveis pode ser uma solução viável para enriquecer a experiência do usuário no processo de compras pela Internet,

dando a possibilidade de visualizar o produto em diversos ambientes reais antes mesmo de finalizar a compra.

### **1.1 O PROBLEMA E SUA IMPORTÂNCIA**

Ao confrontar as mais diversas opções de produtos com suas diversas variáveis como cor, tamanho, espaço do local destinado, harmonização com outros objetos, entre outras peculiaridades, o usuário pode ter dificuldade na tomada de decisão de qual produto comprar, visto que atualmente as compras por lojas virtuais não possibilitam uma pré-visualização do produto no ambiente do cliente. Dessa maneira, muitos optam por realizar a compra de determinado produto na loja física ao invés da loja virtual.

Uma solução para o problema exposto é a aplicação da RA, já existente em diversas áreas como medicina, marketing, engenharia e jogos digitais. Algumas empresas já tem disponibilizado aplicativos para dispositivos móveis com uso de RA, entretanto, grande parte do uso comercial de RA está focada no marketing e são direcionados a um catálogo específico de cada empresa, deixando o usuário restrito a esta empresa (SAÇASHIMA, 2011) (FACHINETTO, 2013).

Se uma loja virtual disponibiliza, além de imagens e vídeos, o modelo tridimensional do objeto, a tecnologia de RA em dispositivos móveis pode se tornar uma ferramenta de auxílio no processo de compra de produtos pela Internet, já que essa ferramenta é um avanço nas maneiras tradicionais de visualização, permitindo que o consumidor conheça melhor o produto que está comprando e o visualize em seu próprio ambiente real antes mesmo de finalizar a compra.

Dessa maneira o uso de RA nas vendas pela Internet pode permitir o usuário ter uma melhor noção do produto que está comprando, dado o alto grau de interação que aproxima o produto do cliente e possibilita melhor tomada de decisão na escolha do produto (FACHINETTO, 2013).

## **1.2 OBJETIVOS**

A seguir são descritos os objetivos gerais e específicos do presente trabalho.

### **1.2.1 Objetivo geral**

Avaliar a aplicação da Realidade Aumentada por vídeo baseado em monitor, sem o uso de marcadores, utilizando um smartphone como dispositivo de captura e visualização, no processo de compras de produtos pela Internet.

### **1.2.2 Objetivo específicos**

1. Construir uma página de vendas de produtos.
2. Obter os objetos virtuais para disponibilizar na página de vendas.
3. Implementar uma aplicação móvel para acessar a página de venda e fornecer a visualização dos objetos virtuais com RA.
4. Disponibilizar a aplicação na loja do Android.
5. Apresentar a aplicação aos usuários.
6. Realizar uma avaliação qualitativa sobre a utilização da RA.

## **1.3 ESTRUTURA DO TRABALHO**

O trabalho está organizado da seguinte forma. A seção 2 traz uma revisão da literatura e é subdividido em 4 sub-seções organizados como segue: a subseção 2.1 traz um breve histórico e características da Realidade Virtual (RV), pois a RA é uma subárea da RV; na subseção 2.2, a RA é abordada extensivamente trazendo definições e classificações dos diferentes tipos de sistemas e técnicas utilizadas; a subseção 2.3 trata do uso da RA mais especificamente com dispositivos móveis; e na subseção 2.4, aborda-se sobre modelagem 3D, já que a aplicação fará uso de objetos tridimensionais. A seção 3 detalha a metodologia utilizada. Na seção 4 apresenta a análise dos resultados. Por fim, na seção 5 é apresentado a conclusão deste trabalho.

## **2 REVISÃO DA LITERATURA**

A Realidade Aumentada só se tornou viável pelo avanço das técnicas de multimídia e da Realidade Virtual, o avanço computacional possibilitou a integração do vídeo em tempo real e ambientes virtuais de forma interativa, o que antes ficava restrito a computadores de grande porte e dispositivos sensoriais foi expandido para microcomputadores e dispositivos móveis (TORI e KIRNER, 2006; KIRNER e KIRNER, 2011). Logo, a RA emergiu como uma subárea de pesquisa da Realidade Virtual, e por esse motivo, é necessário retratar o conceito e histórico de evolução da Realidade Virtual para entender a RA.

### **2.1 REALIDADE VIRTUAL**

A humanidade sempre buscou diferentes maneiras de representar a realidade ou a imaginação, seja através de pinturas, teatro, cinemas e mais recentemente jogos digitais. O avanço da capacidade computacional possibilitou a criação de ambientes totalmente digitais permitindo uma potente forma de representação (TORI e KIRNER, 2006; SAÇASHIMA, 2011).

A Realidade Virtual é uma interface avançada para uma aplicação computacional, pois o usuário pode visualizar, movimentar-se e interagir em ambientes tridimensionais e seus elementos em tempo real, utilizando dispositivos multissensoriais, além disso essa aplicação pode ser enriquecida para além da visualização e permitir um estímulo de sentidos como tato e audição (KIRNER; PINHO, 1997); TORI; KIRNER, 2006).

Segundo Tori e Kirner (2006), a modelagem de ambientes virtuais permite ao usuário visualizar e interagir com objetos e outros elemento presentes no cenário virtual, sendo que uma das formas de interação mais simples consiste na navegação, isto é, a movimentação do usuário pelo espaço tridimensional como um passeio exploratório. Objetos virtuais podem ser animados e possuem comportamentos programados, já que todo o ambiente virtual é criado pelo computador.

Dado o conceito, o foco da Realidade Virtual está na imersão do usuário com um ambiente tridimensional, permitindo a visualização de um ambiente virtual em tempo real e sua interação com o mesmo. Segundo Saçashima (2011) a imersão em sistemas de Realidade Virtual é desejada, devido a ampliação dos sentidos do indivíduo (a possibilidade de interagir em ambientes criados artificialmente) e a disponibilidade de interfaces mais naturais, usando movimentos corporais para navegar e acionar comandos nesses ambientes, por exemplo.

Essa interação é suportada de diversas maneiras, como interação por gestos corporais, comandos de voz, mouse, teclado e o uso de dispositivos não convencionais como luvas e capacetes, desde que o usuário tenha a impressão de estar de fato em um ambiente virtual interagindo com ações naturais como apontar, pegar, manipular e outras ações, todas elas em tempo real.

Para que a aplicação dê a sensação de ocorrer em tempo real, é necessário que o atraso entre a ação executada pelo usuário e resultado obtido no ambiente virtual não ultrapasse 100 milissegundos, dessa maneira um sistema de RV impõe requisitos de hardware e software para que o sistema funcione em taxas de quadro aceitáveis e com menor atraso nas reações aos comandos do usuário (TORI; KIRNER 2006).

### **2.1.1 Breve Histórico**

Apesar da Realidade Virtual ser relacionada com tecnologia computacional de ponta, veremos que essa não é uma área de pesquisa tão recente. Antes mesmo do surgimento do termo RV já haviam pesquisas e experimentos a fim propiciar ao usuários uma imersão em um mundo virtual. Na década de 1950, o primeiro dispositivo a propiciar a imersão do usuário em um ambiente virtual tridimensional foi criado. Na década de 1960 um engenheiro criou o primeiro capacete de Realidade Virtual e, somente na década de 1980 um artista e cientista da computação cunhou o termo Realidade Virtual (TORI; KIRNER 2006). Essas etapas são melhores descritas a seguir.

Como pioneiro no desenvolvimento de um sistema imersivo, temos Morton Heilig que na década de 1950 propôs um simulador de imersão denominado Sensorama.

O sistema proporciona visão estereoscópica, sons estéreos, odores, ventos e vibrações, produzindo assim a ilusão de um passeio de moto, conforme demonstrado na Figura 1 (PIMENTEL e TEIXEIRA, 1995; PARENTE, 1999).



Figura 1 - Sensorama.  
Fonte: Fachinetto (2013)

O primeiro HMD<sup>1</sup> funcional foi construído em 1961 pela Philco Corporation, porém ele é mais considerado uma tecnologia de RA do que de RV. Segundo D'Angelo (2016) o display fornecia ao piloto uma visão aumentada através de captura de uma câmera infravermelha fixada na parte inferior do helicóptero usado para auxiliar o piloto em pouso noturno. Como a imagem é aumentada, este foi considerado o primeiro sistema de RA por vídeo.

Segundo OLIVEIRA (2008) o pioneiro da Realidade Virtual é Ivan Sutherland que também foi responsável por originar o termo conhecido como Computação Gráfica, após criar o SketchPad em 1963 no *Massachusetts Institute of Technology* (MIT). O pesquisador passou a realizar pesquisas com diversos tipos de capacetes e ao final da década de 60 criou o “*The Sword of Damocles*”, considerado o primeiro HMD com capacidade de rastreamento de cabeça e sobreposição em tempo real de imagem gerada por computador, o dispositivo pode ser conferido na Figura 2 a seguir.

---

<sup>1</sup> HMD (Head-Mounted-Display): A tradução direta seria “display montando na cabeça”, geralmente usado como um capacete que possui um display que fica em frente aos olhos do usuário.

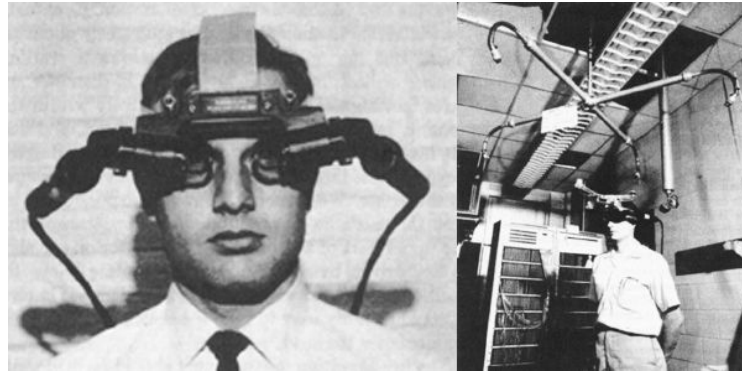


Figura 2 - Sword of Damocles.  
Fonte: D'Angelo (2017)

Segundo Saçashima (2011, apud Krueger, 2004) em 1973 Myron Krueger cunhou o termo Realidade Artificial para descrever seu sistema Videoplace, considerado por alguns autores como pioneiro na Realidade Virtual, o sistema criado por Krueger consistia em uma combinação de computadores e sistemas de vídeo e respondia aos movimentos do observador permitindo interações sem a necessidade de luvas ou capacetes, conforme a Figura 3. Além disso esse ambiente era compartilhado permitindo interação com outras pessoas.



Figura 3 - Videoplace por Myron Krueger.  
Fonte: Young(2010)

Somente no final da década de 1980 que Jaron Lanier cunhou o termo Realidade Virtual, o artista e cientista propôs a união de dois conceitos opostos (Realidade e Virtual) em um novo conceito que capta a essência dessa tecnologia que é justamente a fusão do real com o virtual.



### 2.1.2 Classificação dos sistemas de Realidade Virtual

Segundo Jacobson (1994), os sistemas de RV se diferem de acordo com o nível de imersão e interação que proporcionam ao usuário, esses níveis de imersão e interação são determinados conforme os dispositivos de entrada e saída utilizados no sistema de RV.

Uma proposta de classificação dos diferentes sistemas de RV é dado por Pimentel e Teixeira (1995), a seguir:

- RV de simulação;
- RV de projeção;
- RV de mesa;
- Telepresença;
- Displays Visualmente Casados (*Visually Coupled Display*);
- Realidade Aumentada (*Augmented Reality*).

**RV de simulação:** Sistemas de simulação de Realidade Virtual geralmente são voltados para desenvolvimento e treinamento, conforme a Figura 4. A ideia é construir uma cabine com controles e telas que tentam imitar o interior de um veículo, por exemplo, e apresentar um mundo virtual que reaja a esses controles. As características desse tipo de RV é ter resposta dinâmica e fiel, de forma a representar completamente o sistema simulado ( II Encontro de Ferrovias, 2010).



Figura 4 - Simulador de trem desenvolvido pela USP para a empresa VALE S.A.  
Fonte: USP (2017)

**RV de projeção:** Segundo Jacobson (1994), no sistema de RV de projeção, o usuário está fora do mundo virtual mas pode interagir com personagens e objetos contidos nele. Esse tipo de RV também é conhecido como Realidade Artificial e foi criada nos anos 70 por Myron Krueger com seu sistema VIDEOPLACE, um ambiente que capturava a imagem dos usuários e respondia aos movimentos do observador, podendo este interagir com outros usuários ou objetos (NETTO; MACHADO; OLIVEIRA, 2002). A Figura 5 traz um exemplo de sistema de RV de projeção.



Figura 5 - Mesa de projeção utilizada na copa do mundo 2014.  
Fonte: Pereira e Nogueira (2017)

**RV de mesa:** O sistema de Realidade Virtual de mesa é similar ao sistema tradicional de Realidade Virtual, com a diferença no dispositivo que é usado para a imersão. No sistema tradicional temos o uso de *head mounted displays* ao passo que no sistema de RV de mesa são utilizados grandes monitores (através de telas ou projeção) para apresentação do mundo virtual (NETTO; MACHADO; OLIVEIRA, 2002).

**Telepresença:** Esse tipo de sistema se caracteriza pela utilização de câmeras de vídeo e microfones a fim de projetar o usuário no mundo virtual, sistemas de telepresença tem sido alvo de muitas pesquisas, principalmente para uso na medicina. Pois através de câmeras o médico pode ter uma visão interna do paciente (NETTO; MACHADO; OLIVEIRA, 2002).

**Displays Visualmente Casados:** Segundo Pimentel e Teixeira (1995), essa é a classe mais presente em sistemas de RV. Em sistemas de displays visualmente

casados, as imagens são exibidas para o usuário através de um dispositivo que deve acompanhar os movimentos de sua cabeça. Esses dispositivos permitem tanto imagens quanto sons estéreos e sensores que detectam a movimentação da cabeça do usuário. Através desses movimentos o sistema é alimentado com informações que mantêm alinhado os sons e imagens conforme a perspectiva do usuário.

**Realidade Aumentada:** Ao passo que a Realidade Virtual é aplicada normalmente em ambientes fechados com a dependência de equipamentos de visualização como os já citados monitores, projetores, HMD e sensores, a RA surge como uma alternativa mais abrangente pois pode ser aplicada em qualquer ambiente. Na RA o usuário não deixa de enxergar o mundo real, logo o mesmo passa a visualizar objetos virtuais como: animações, dados e gráficos 3D sobrepostos ao mundo real (NETTO; MACHADO; OLIVEIRA, 2002; FACHINETTO, 2013).

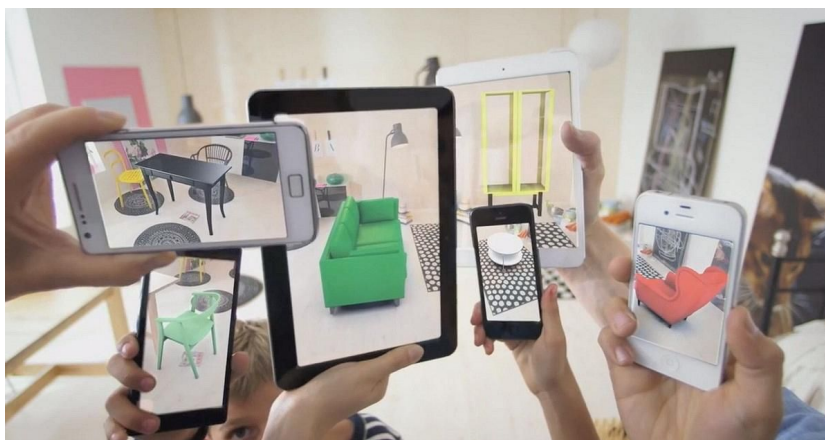


Figura 6 - Realidade Aumentada em dispositivos móveis.  
Fonte: Ramos (2016)

Por se tratar do foco do presente trabalho, a RA será abordada de forma mais abrangente na próxima Subseção. Os outros tipos de sistemas de RV assim como as técnicas utilizadas por estes não são foco deste trabalho. Para maiores detalhes a respeito consultar (STEUER, 1992; NETTO; MACHADO; OLIVEIRA, 2002; TORI; KIRNER, 2006).

## 2.2 REALIDADE AUMENTADA

O termo Realidade Aumentada (RA) foi definido nos anos 90 pelo pesquisador Thomas Caudell, da Boeing. Na época o termo foi aplicado à um sistema que orientava os profissionais no momento da montagem de cabos elétricos da aeronave, utilizando um HMD.

O avanço dos sistemas de Realidade Virtual e avanço do poder computacional dos dispositivos permitiu integrar em tempo real vídeo com ambientes virtuais interativos. A RA se beneficiou desse progresso, tornando viáveis aplicações com essa tecnologia tanto em plataformas sofisticadas quanto em plataformas mais populares (TORI; KIRNER, 2006).

Ainda segundo Tori e Kirner (2006), enquanto a Realidade Virtual transporta o usuário para o ambiente virtual, a RA transporta o ambiente virtual para o espaço físico do usuário, essa inversão possibilita uma interação com o mundo virtual de maneira mais natural dispensando etapas de treinamento e adaptação.

Azuma et al. (2001) define RA simplesmente como a adição de elementos virtuais em tempo real ao mundo real, de forma que os elementos virtuais parecem fazer parte do ambiente físico. A RA faz parte de uma subárea da Realidade Virtual chamada realidade mista ou realidade misturada que é justamente o que está entre o mundo real e um mundo totalmente virtual. Milgram (1994) define Realidade-Virtualidade contínua como a conexão do ambiente real para um ambiente totalmente virtual, conforme exibido na Figura 7.



Figura 7 - Realidade-Virtualidade contínua, por Paul Milgram, traduzido por Saçashima.  
Fonte: Saçashima (2011)

Sendo o ambiente real o mundo que nós conhecemos, a RA é o ambiente em que há uma predominância do mundo real sobreposto por objetos virtuais, já a virtualidade aumentada é a predominância do virtual no qual é inserido objetos reais, e o ambiente virtual é aquele totalmente modelado por computadores, conforme demonstrado Figura 8.



Figura 8 - Demonstração da Realidade-Virtualidade Contínua.  
Fonte: Adaptado de Duffy (2016).

Saçashima (2011) afirma que um sistema de RA tem caráter multidisciplinar, agregando diversos conhecimentos e tecnologias de diferentes disciplinas, dentre estas a de computação gráfica e visão computacional, esta última de grande importância para as técnicas de rastreamento baseado em vídeo, que basicamente calcula a posição correta do modelo 3D no mundo real.

### 2.2.1 Classificação dos sistemas de Realidade Aumentada

Azuma et al. (2001) propõe uma classificação para os variados tipos de sistemas de RA, essa classificação varia de acordo com o dispositivo usado para visualização:

- **Sistema de visão ótica direta** (*optical see-through*): Utiliza um visor semi transparente como óculos ou capacete que permite o utilizador enxergar o mundo real ao mesmo tempo que as informações virtuais são sobrepostas.
- **Sistema de visão direta por vídeo** (*video see-through*): também utiliza óculos ou capacete, a diferença é que o usuário não enxerga o ambiente real com os próprios olhos mas sim através de câmeras. A imagem do ambiente



real é capturada e processada adicionando elementos virtuais. Assim a imagem final é exibida para o usuário através de monitores.

- **Sistema de visão ótica por projeção** (*projection-based augmented reality*): Esse tipo de sistema projeta os objetos virtuais diretamente na superfície do ambiente real, eliminando o uso de visores como óculos, capacetes e monitores. Todavia o sistema apresenta outras limitações como a dependência da superfície a ser projetada e o correto alinhamento dos projetores.
- **Sistema de visão por vídeo baseado em monitor** (*monitor-based augmented reality*): Nesse sistema uma câmera captura a cena real, que é processada a fim de acrescentar os objetos virtuais e logo após exibir em um monitor. O usuário passa a enxergar a RA através do monitor, como exemplo apresentado na Figura 9. Por isso esse sistema também pode ser classificado como visualização indireta. Os smartphones se encaixam nessa categoria fazendo com que esse tipo de sistema se torne extremamente popular, devido o baixo custo e fácil acesso.



Figura 9 - Aplicações de Realidade Aumentada baseada em monitor.  
Fonte: Parkin (2016)

A Figura 10 a seguir apresenta uma síntese das vantagens e desvantagens de cada sistema.

Sistema	Vantagens	Desvantagens
Visão Ótica Direta	<ul style="list-style-type: none"> <li>- Não elimina a visão direta do mundo real, diminuindo a gravidade em caso de pane no sistema;</li> <li>- Facilita a mobilidade;</li> <li>- Não limita o campo de visão do ambiente real;</li> </ul>	<ul style="list-style-type: none"> <li>- Dificuldades de equalização de brilho e contraste, para uma integração visual adequada dos elementos virtuais com o ambiente real;</li> <li>- Não permite a oclusão de objetos reais pelos virtuais, dificultando a distinção entre ambos;</li> <li>- Necessita a acoplagem de dispositivos pelo corpo;</li> </ul>
Visão Direta Por Vídeo	<ul style="list-style-type: none"> <li>- Facilita a equalização de brilho, contraste e resolução;</li> <li>- Permite a oclusão de objetos reais pelos virtuais, melhorando a distinção entre ambos;</li> <li>- Facilita a mobilidade, exceto em caso de uso de capacete HMD;</li> </ul>	<ul style="list-style-type: none"> <li>- Campo de visão limitado pelo tamanho do vídeo;</li> <li>- No caso do uso de capacete HMD elimina a visão direta do mundo real e a visualização é comprometida em caso de pane no sistema. Além da necessita a acoplagem do dispositivo no corpo;</li> </ul>
Visão Ótica Por Projeção	<ul style="list-style-type: none"> <li>- Não elimina a visão direta do mundo real, diminuindo a gravidade em caso de pane no sistema;</li> <li>- Dispensa a acoplagem de dispositivos pelo corpo;</li> </ul>	<ul style="list-style-type: none"> <li>- Necessidade de modelar previamente os objetos da cena;</li> <li>- A iluminação do ambiente e as cores e texturas das superfícies interferem na projeção;</li> <li>- Mobilidade limitada pela projeção;</li> </ul>
Visão Por Vídeo Baseado Em Monitor	<ul style="list-style-type: none"> <li>- Baixo custo;</li> <li>- Dispensa a acoplagem de dispositivos pelo corpo;</li> <li>- Permite a oclusão de objetos reais pelos virtuais, melhorando a distinção entre ambos;</li> <li>- Facilita a equalização de brilho, contraste e resolução;</li> </ul>	<ul style="list-style-type: none"> <li>- Não possibilita a visão direta do ambiente;</li> <li>- Campo de visão limitado;</li> <li>- Mobilidade limitada;</li> </ul>

Figura 10: Vantagens e desvantagens entre os sistemas de Realidade Aumentada.  
Fonte: Fachinetti (2013)

### 2.2.2 Classificação das técnicas de interação

A Interação é um tipo de ação que ocorre entre duas ou mais entidades quando a ação de uma delas provoca uma reação da outra ou das restantes. Segundo Bastos, Teichrieb e Kelner (2006) a interação tanto em sistemas de RV quanto de RA é um tópico muito explorado, pois busca possibilitar ao usuário uma forma mais natural de interagir com o sistema. Um aspecto importante é permitir a interação simultânea dos objetos reais e virtuais.

Dada as características da RA, existem diferentes formas de interação com uma aplicação, interação esta que pode ser tanto com um objeto real quanto um objeto virtual. Broll et al. (2005) propõe a seguinte classificação das técnicas de interação:

- **Interação Espacial** (*spatial interaction*): Este tipo de interação consiste em manipular objetos físicos, esses objetos são mapeados pelo sistema de RA e conforme são manipulados ocorre uma transformação nos objetos virtuais, de forma natural e intuitiva.
- **Interação Baseada em Comandos** (*command-based interaction*): Esse tipo de interação se dá através de comandos, seja por gestos, voz ou através de sensores que determinam o posicionamento, como por exemplo Playstation Move. Com a evolução do reconhecimento por voz, esse tipo de interação tem se tornado bastante popular no uso de diversos sistemas, incluindo sistemas de RA.
- **Interação por Controle Físico** (*physical control interaction*): Interação por controle físico é muito utilizada em ambientes de simulação. Um volante real pode ser usado para controlar um objeto virtual em um ambiente de simulação de veículos, por exemplo.
- **Interação por Controle Virtual** (*virtual control interaction*): O próprio controle é virtualizado. Ao manipular esses controles virtuais alguma função pré programada é executada. Um exemplo são os controles virtuais que aparecem em jogos de smartphones, sendo esta inclusive o tipo de interação que será desenvolvida neste trabalho.

### 2.2.3 Técnicas de rastreamento

Devido ao baixo custo e disponibilidade, grande parte das pesquisas de RA está ligada a captura por vídeo, ou seja, ao filmar o ambiente real é realizado um processamento das imagens<sup>2</sup> a fim de inserir os objetos virtuais em tempo real. Assim, ao conferir a imagem processada através de um monitor, HMD ou tela do smartphone o usuário irá ver a RA. Para manter essa continuidade da forma mais natural possível é necessário calcular precisamente a posição do objeto virtual no ambiente real.

---

<sup>2</sup> Um vídeo é uma sucessão de imagens que passam a impressão de movimento.



Esse correto posicionamento 3D dos objetos virtuais é chamado de “Registro”, tal tarefa é possível através do uso de técnicas de rastreamento (*tracking*). Essas técnicas usam visão computacional para perceber características do ambiente real e, a partir dessas informações, calcular o posicionamento do objeto virtual em relação a posição do observador. As informações necessárias para aplicar tais técnicas são obtidas através de vários sensores como vídeo e GPS (LIMA, 2010).

Existem duas categorias de objetos que são rastreados pelo sistema de RA, são eles:

1. Marcadores Fiduciais (rastreamento com marcadores);
2. Informações naturais da cena (rastreamento sem marcadores).

#### **2.2.3.1 Rastreamento com marcadores**

Os marcadores fiduciais é uma das maneiras mais simples e menos custosa computacionalmente de se fazer o rastreio em uma cena. Esses marcadores são imagens impressas em preto e branco em superfície plana (geralmente uma folha de papel), similar ao QR Code<sup>3</sup> exibido na Figura 11, e funcionam como um ponto de referência o qual será sobreposto por um objeto virtual. Segundo Saçashima (2011) a aplicação é capaz de reconhecer e processar tais padrões, calculando assim o seu posicionamento em relação às imagens obtidas pela câmera. Apesar do processamento ser mais simples e menos custoso computacionalmente, esse tipo de rastreio apresenta o inconveniente de ser intrusivo e não natural, uma vez que os marcadores devem estar sempre presentes na cena.

---

<sup>3</sup> QR Code é um código de barras bidimensional que pode ser facilmente escaneado usando a maioria dos telefones celulares equipados com câmera.

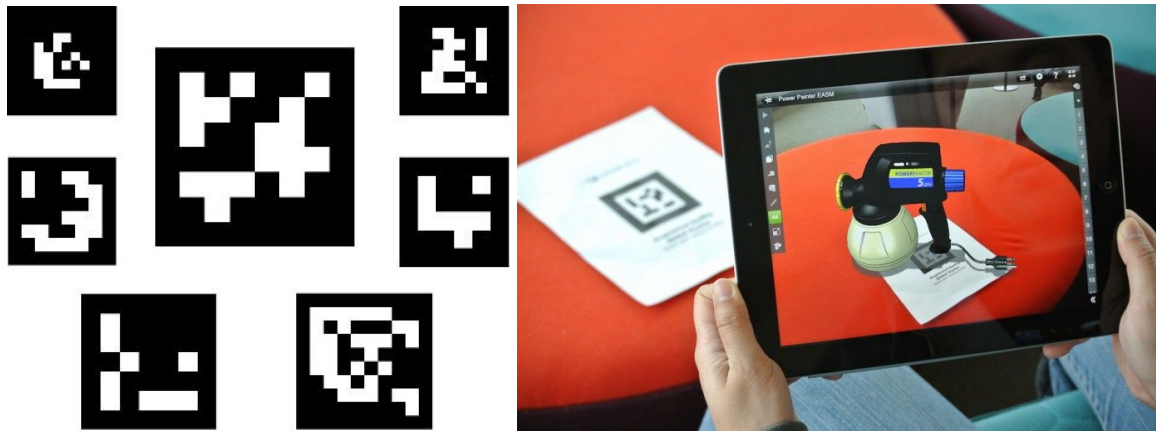


Figura 11 - Exemplos de marcadores fiduciais.  
 Fonte: OpenCV (2017) e Multidots (2017)

### 2.2.3.2 Rastreamento sem marcadores

Essa abordagem não utiliza os marcadores fiduciais, ao invés disso o sistema percebe características presentes na cena real como contornos e texturas, os contornos consistem na silhueta dos objetos, assim como suas arestas internas, já as texturas considera características dos objetos tal como suavidade, rugosidade e brilho. Essas características passam a ser usadas como marcadores servindo de ponto de referência para posicionar os objetos virtuais (LIMA, 2010; LIMA et al., 2015).

Existem diversas técnicas de rastreamento sem marcadores e estas podem ser classificadas em dois tipos:

1. Baseada em modelo
2. Sem modelo / Baseada em reconstrução 3D (Lima et al., 2007).

Nas técnicas baseadas em modelo, é preciso conhecer o mundo real previamente, esse conhecimento é armazenado em um modelo 3D e utilizado posteriormente para recuperar a pose da câmera (SIMÕES, 2011; SAÇASHIMA, 2011). Já nas técnicas baseada em reconstrução 3D, a geometria da cena assim como o posicionamento da câmera são estimados sem nenhum conhecimento prévio, sendo estes adquiridos durante o rastreamento, fazendo deste um processo mais natural, porém mais custoso e complexo.

As técnicas baseadas em modelo podem ser divididas em duas categorias (LIMA, 2010; SIMÕES, 2011):

- **Rastreamento recursivo:** Nesta técnica a pose anterior da câmera é usada na estimativa da pose atual.
- **Rastreamento por detecção:** A pose da câmera é calculada sem nenhuma estimativa prévia.

Simões (2011) apresenta na Figura 12 uma taxonomia para as técnicas sem marcadores baseada em modelos.

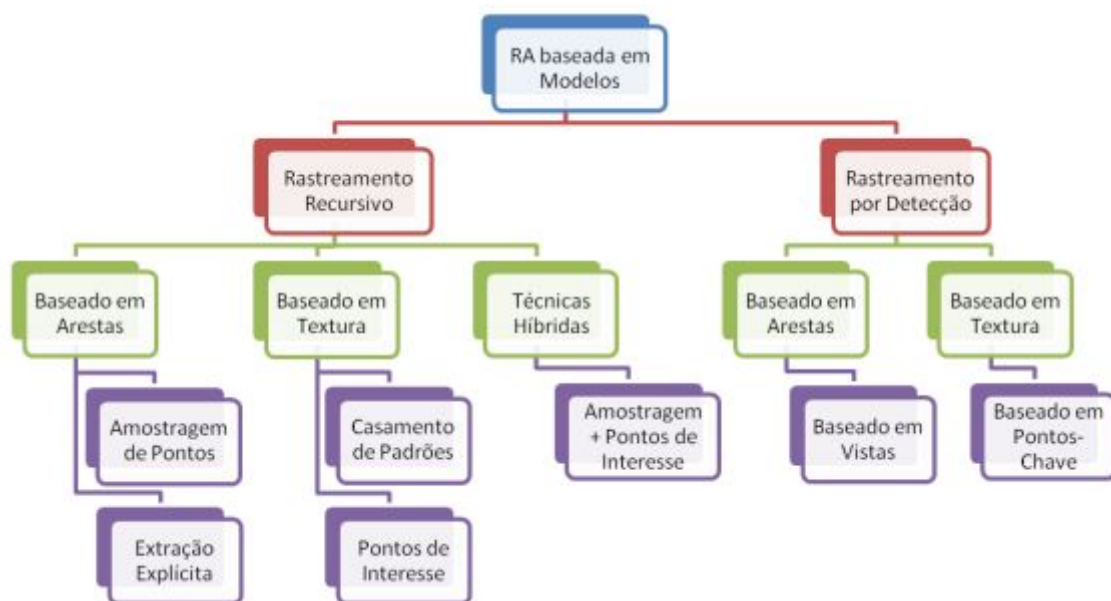


Figura 12 - Taxonomia das técnicas sem marcadores baseadas em modelos.  
Fonte: Simões (2011)

Observe na Figura 12 que as técnicas também podem ser classificadas em três categorias conforme o tipo de característica utilizada no rastreamento (LIMA et al., 2010):

1. **Baseado em arestas:** A pose da câmera é estimada ao casar o modelo 3D com as informações de arestas extraídas da imagem.
2. **Baseado em texturas:** Similar ao uso de marcadores, com o detalhe que os “marcadores” passam a ser texturas presentes na cena, como a capa de um livro por exemplo.

**3. Técnicas híbridas:** Utilizam informações tanto de arestas quanto de texturas de maneira complementar.

A Figura 13 traz uma síntese dos métodos baseado em modelos, quanto a processamento, precisão e robustez.

<b>Categoria</b>	<b>Método</b>	<b>Deteccção</b>	<b>Processamento</b>	<b>Precisão</b>	<b>Robustez</b>
Rastreamento recursivo	Baseado em aresta	Não	Baixo	<i>Jitter</i>	Sensível a: • Movimento rápido de câmera • Fundo poluído
	Baseado em fluxo óptico	Não	Baixo	Erros cumulativos	Sensível a: • Movimento rápido de câmera • Mudanças de iluminação
	Casamento de padrão	Não	Baixo	Altamente preciso	Sensível a: • Movimento rápido de câmera • Mudanças de iluminação • Oclusão
	Baseado em ponto de interesse	Não	Alto	Preciso	Sensível a: • Movimento rápido de câmera
Rastreamento por deteção	Baseado em vistas	Sim	Alto	Preciso	Intervalo restrito de poses
	Baseado em pontos-chave	Sim	Alto	<i>Jitter e drift</i>	Robusto

Figura 13 - Comparação das técnicas de rastreamento baseado em modelo.  
Fonte: Lima (2010)

Os trabalhos de Lima (2010) e Simões (2010) abordam mais detalhadamente as técnicas baseada em modelos.

### **Rastreamento sem marcadores baseados em reconstrução 3D**

Como já foi dito, técnicas baseadas em reconstrução 3D não necessitam de informações prévias da cena, essas técnicas são divididas em duas categorias:

1. *Structure from Motion* (SfM);
2. *Simultaneous Localization and Mapping* (SLAM).

Nos métodos baseados em SfM, o rastreamento e recuperação da estrutura da cena acontece em tempo real, com diferentes níveis de detalhes variando conforme a

abordagem usada. A recuperação da cena ocorre a partir de informações da geometria epipolar<sup>4</sup> (NISTÉR, 2003; LIMA, 2010; SIMÕES, 2011). Segundo Lima (2010), as técnicas baseadas em SfM não necessitam de uma etapa de treinamento, possibilitando a reconstrução de um ambiente totalmente desconhecido, todavia, isso torna as abordagens baseadas em SfM bastante complexas. SfM é uma técnica oriunda da visão computacional a qual possibilita tanto o rastreo de uma cena quanto aquisição da estrutura presente nessa cena (SIMÕES, 2011; POLLEFEYS, 1999, apud LIMA, 2010). Segundo Pollefeys et al. (2007), SfM tem sido largamente utilizado para reconstrução 3D a partir de imagens, conforme exibido na Figura 14.



Figura 14 - Bundler: Reconstrução 3D a partir de uma coleção de imagens não ordenadas.  
Fonte: Noah, Seitz e Szeliski (2006)

A implementação tradicional de SfM sugere um *pipeline* constituído das seguintes fases (LIMA, 2010; SIMÕES, 2011):

- **Rastreamento de características:** identifica e relaciona os pontos de interesse que podem ser detectados em várias imagens.
- **Geração de hipóteses de câmera:** Calcular as posições da câmera através dos conceitos de geometria epipolar.
- **Auto-calibração:** Ajuste dos parâmetros internos da câmera
- **Triangulação:** Obtenção dos pontos 3D
- **Estimação de mapas de profundidade densos e geração do modelo 3D texturizado:** Estima a posição dos pontos 3D de acordo com a profundidade, e através da texturização aplica informações de cor a esses pontos de forma a reconstruir fielmente o objeto.

<sup>4</sup> Geometria epipolar é a geometria da visão estéreo. Quando duas câmaras veem uma cena 3D a partir de duas posições distintas.

Olsson (2012) explica o *pipeline* de SfM de forma simplificada, em quatro passos:

1. Obter uma grande quantidade de imagens de um determinado objeto em diferentes ângulos de visão (Figura 15A e Figura 15B).
2. Identificar pontos em comuns nessas imagens como arestas e cantos (Figura 15C e Figura 15D).
3. Procurar por pontos correspondentes aos identificados anteriormente em outras imagens (Figura 15E e Figura 15F).
4. Computar as posições da câmera e dos pontos 3D de forma que os raios de visão correspondentes se cruzem (Figura 15G).

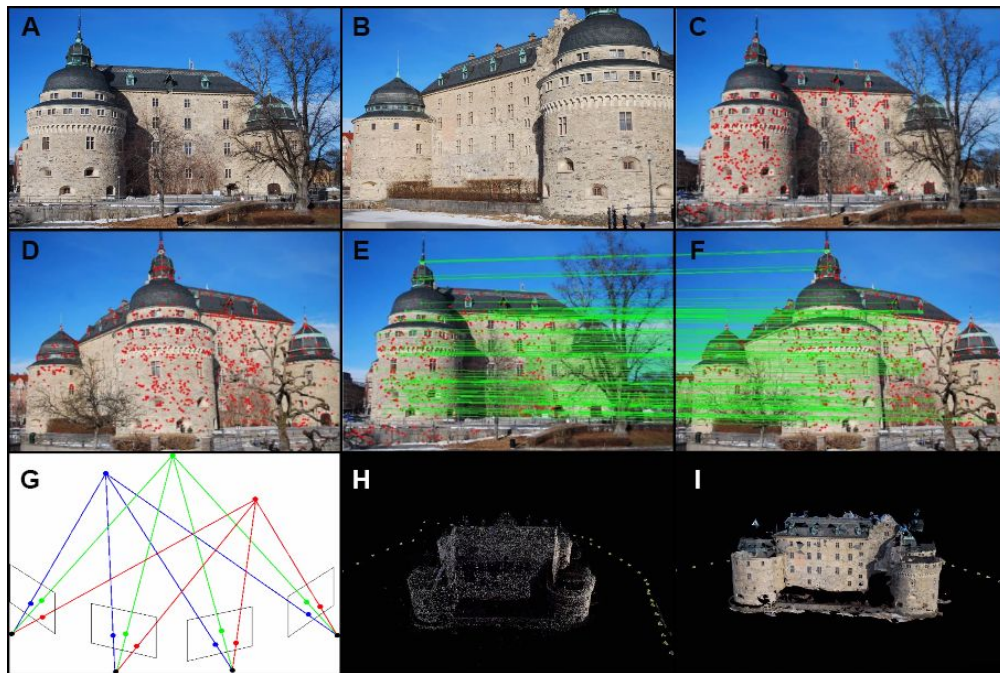


Figura 15 - *Pipeline* do SfM, o resultado final em (H) foi obtido processando 671 imagens do castelo, o que gerou aproximadamente 50000 pontos 3D, em (I) temos a superfície computada a partir dos pontos 3D já aplicando a textura do castelo.

Fonte: Olsson (2012)

O *pipeline* tradicional de SfM não restringe a técnica para funcionamento em tempo real, todavia alguns algoritmos presentes no *pipeline* requisitam um grande tempo de processamento, inviabilizando assim o uso em tempo real, logo, é necessário realizar otimizações no *pipeline* do SfM caso a aplicação necessite funcionar de forma online (LIMA, 2010).

A fim de otimizar o *pipeline* do SfM para viabilizar o rastreamento em tempo real, algumas etapas são eliminadas, como a auto-calibração e texturização, pois nessa abordagem o rastreo é mais importante e não a reconstrução 3D completa. Diversas etapas são simplificadas e passam a utilizar o poder de processamento da GPU (SIMÕES, 2011).

Em Newcombe e Davison (2010) temos um belo exemplo de otimização do *pipeline* do SfM, em que é demonstrado uma reconstrução completa da cena em tempo real (reconstrução densa) com alta taxa de precisão e RA avançada com simulação de física onde os objetos virtuais interagem com objetos reais, tudo utilizando apenas uma câmera.

Já a técnica de SLAM (*Simultaneous Localization and Mapping*) é uma abordagem muito utilizada na área de robótica para identificar obstáculos e permitir uma navegação autônoma de robôs, pois a técnica consiste em construir (não completamente) um ambiente desconhecido ao mesmo tempo que guarda a posição de um agente dentro do mapa, estimando tanto a pose da câmera quanto a geometria 3D da cena em tempo real.

Em sua abordagem tradicional o SLAM utiliza além de câmeras, sensores como scanners a laser e sonares, já a abordagem de SLAM Monocular utiliza apenas uma câmera como sensor. Ao contrário das técnicas de SfM, as técnicas de SLAM trabalham com a reconstrução esparsa (quantidade de pontos inferior à quantidade de pixels das imagens) da cena, possibilitando desta forma um rastreo em tempo real com foco na localização e não na reconstrução completa da cena (LIMA, 2010), (SIMÕES, 2011) e (LIMA et al, 2015).

Os algoritmos presentes nas técnicas de SLAM detectam marcos visuais, isto é, pontos de interesse nas cenas, esses pontos podem ser complementados através do uso de descritores. Esses algoritmos assumem que os pontos de interesse são projeções de pontos no espaço tridimensional e ao obter um conjunto de pontos, em diferentes imagens, é estimado a posição desses pontos no espaço assim como o movimento da câmera. Além de pontos, informações da cena podem ser obtidas de curvas e retas e pode ser de grande valia para ambientes internos onde a própria



estrutura é composta por diversas retas, conforme exibido na Figura 16 (WERNECK e COSTA, 2010).

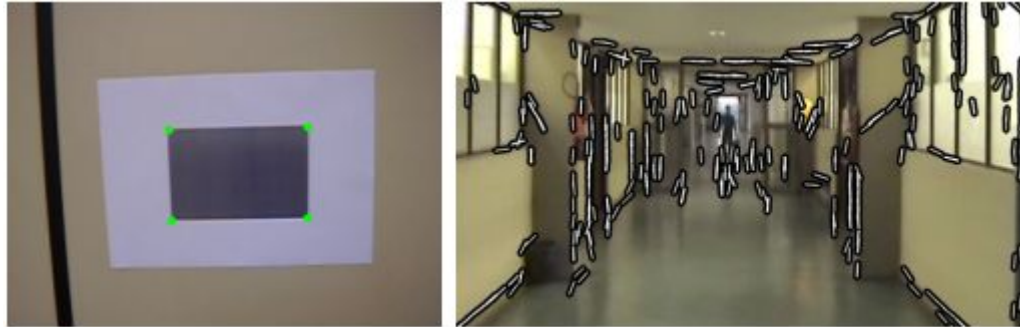


Figura 16 - SLAM: detecção de marcos pontuais (esquerda) e retas (direita).  
Fonte: Werneck e Costa (2011).

Werneck e Costa (2010) afirmam que apesar dos resultados obtidos com SLAM tradicional serem satisfatórios, eles possuem um elevado custo computacional, que aumenta à medida que o número de marcos aumentam, podendo chegar a complexidade  $O(n^3)$  operações por iteração e a precisão não é a melhor possível.

Segundo Simões (2011):

Para realizar o rastreamento, as técnicas de SLAM utilizam no primeiro quadro da cena um objeto conhecido a uma distância previamente determinada para inicialização do sistema. Isto se dá devido à utilização de um framework de inferência bayesiana sequencial, geralmente através de filtro de Kalman (KF) ou filtro de Kalman estendido (EKF). Através deste framework, utilizando propriedades de previsibilidade, é possível realizar a reconstrução esparsa de marcos (cantos com assinaturas únicas e mais bem definidas que cantos padrão) e rastrear a câmera a uma taxa de 30 quadros por segundo, suficiente para considerar uma aplicação de tempo real.

Uma variação do SLAM é o *Parallel Tracking and Mapping System* (PTAM), onde o rastreamento é de alta densidade baseado em centenas de marcos, ao contrário do SLAM tradicional. Klein e Murray (2009) propõe uma variação do PTAM voltada para smartphones e, após adaptações no PTAM tradicional a implementação para smartphone demonstrou conseguir atingir ótimo desempenho mesmo com o hardware limitado.

Recentemente um grupo de pesquisa brasileiro propôs uma técnica de SLAM Monocular denominada *Simple Tracking and Mapping* (STAM), a qual acabou



vencendo uma competição de rastreamento no IEEE *International Symposium on Mixed and Augmented Reality* (ISMAR) de 2015. Geralmente as câmeras usadas em SLAM Monocular não implementam odometria, logo esta tem que ser feita de forma visual. STAM implementa odometria visual através do rastreamento de características usando fluxo ótico e descritores de características (ARAÚJO, 2016).

### **2.3 REALIDADE AUMENTADA EM DISPOSITIVOS MÓVEIS**

A Realidade Aumentada pode ser implementada em diversas plataformas como Desktop, Web e Móvel. Devido às suas especificidades, cada plataforma tem uma influência direta no grau de complexidade de uma aplicação de RA, sendo que, certas aplicações de RA podem apresentar restrições quanto a mobilidade do usuário. Diante disso as pesquisas de RA com foco em dispositivos móveis tem se tornado cada vez mais populares pois resolvem esse problema da mobilidade (LIMA et al., 2015; ROBERTO; LIMA; TEICHRIEB, 2016).

Apesar da vantagem da mobilidade ao desenvolver aplicações de RA para dispositivos móveis, outras restrições podem surgir relacionadas às limitações de hardware como memória, poder de processamento e espaço de armazenamento. Essas limitações representam um desafio pois vários algoritmos de rastreamento, principalmente os baseados em características naturais, possuem um alto custo computacional (LIMA et al., 2015).

Sistemas de RA em dispositivos móveis são classificados em Distribuídos ou Autônomos. Em sistemas distribuídos parte das tarefas da aplicação de RA são executadas em um servidor, sendo que existem diversos níveis de distribuição. Em sistemas autônomos toda a implementação necessária está apenas no dispositivo móvel, como pode ser observado na Figura 17.

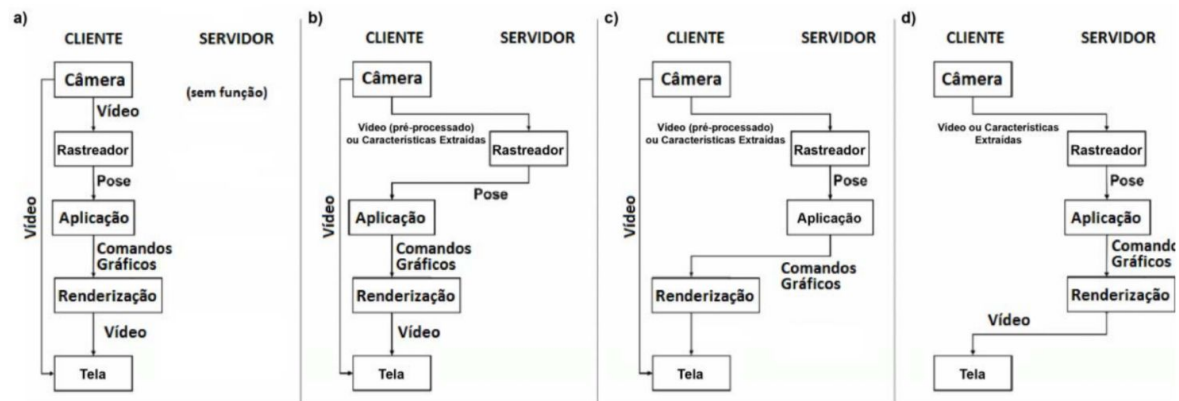


Figura 17 - Arquitetura dos sistemas de RA em dispositivos móveis, em a) temos um sistema autônomo, b) c) e d) temos sistemas com diferentes formas de distribuição de processamento.

Fonte: Lima et al (2015)

Segundo Lima (2010), as soluções autônomas de RA em dispositivos móveis eram limitadas pois a maioria das soluções de RA sem marcadores nesses dispositivos eram distribuídas com a fase de rastreamento ocorrendo no servidor, visto que o servidor pode ter um poder computacional muito maior que um dispositivo móvel. Todavia ao optarmos por uma solução distribuída perdemos a grande vantagem do uso de um dispositivo móvel que é a questão da mobilidade, devido a necessidade de um servidor dedicado e infraestrutura de rede. Outro problema é a questão do atraso e baixa taxa de quadros, já que cada quadro capturado tem que ser enviado para o servidor processar e então retornar uma resposta para a aplicação. Em Lima et al. (2015) temos um comparativo de uma mesma aplicação sendo executada de forma autônoma e distribuída na qual fica claro o problema do atraso de rede.

Lima (2010) ainda afirma, que a maioria das técnicas de rastreamento para dispositivos móveis utilizam marcadores fiduciais, que é um padrão de fácil reconhecimento pela aplicação. Todavia esse método é intrusivo e não desejável em muitos casos.

Contudo, com o avanço do poder de processamento dos dispositivos móveis junto com o avanço das técnicas de rastreamento, tem ocorrido uma mudança no cenário, permitindo que sistemas de RA autônomos com rastreo de características naturais da cena atinjam um ótimo desempenho em dispositivos móveis.

Roberto et al. (2016) realizou um mapeamento sistemático a respeito do rastreamento para dispositivos móveis, em que foram levantadas e classificadas diversas pesquisas publicadas recentemente nessa área. Neste estudo foi verificado que a maioria dos trabalhos consistem em usar uma combinação de sensores do próprio dispositivo (como gps, acelerômetro, wi-fi, giroscópio) para calcular a pose e que, o rastreamento baseado em marcadores, tem sido o método menos utilizado para esta mesma tarefa. Resultados ainda mostram que na maioria dos trabalhos todo o processamento é realizado no dispositivo, isto é, através de soluções autônomas.

Ainda segundo Roberto et al. (2016) diversos trabalhos apresentam uma solução de cálculo de pose 6D, onde são utilizadas informações como a rotação e translação dos dispositivos para calcular a pose, incluindo o cálculo de tamanho (escala) do objeto virtual. Cerca de dois terços das publicações propõem técnicas novas para realizar o rastreo. De uma forma geral fica claro nesse estudo que o número de publicações sobre rastreamento em dispositivos móveis tem aumentado com o passar dos anos e que o rastreo baseado em características naturais tem sido viabilizado devido o avanço computacional desses dispositivos.

De fato, a Realidade Virtual e aumentada em dispositivos móveis tem sido de grande foco também na indústria. O novo chip da Qualcomm Snapdragon 835 para dispositivos móveis conta com alto poder de processamento, menor consumo de energia, GPU dedicada e um processador de sinais que é responsável por reduzir a latência entre o instante que você mexe a cabeça e o dispositivo entende o movimento. A promessa da qualcomm é trazer RV e RA realista para os dispositivos móveis (HIGA, 2017).

Neste contexto, a versão 7 do Sistema Operacional (SO) Android conta com suporte a Realidade Virtual, além disso o SO implementou a Vulkan API para possibilitar gráficos imersivos e com melhor desempenho (POWER, 2016).

A fim de possibilitar experiências de RA mais realistas, a Google lançou o Projeto Tango, esse projeto é uma plataforma para dispositivos móveis que tem como foco

detectar sua posição e o espaço ao redor assim como os seres humanos. O projeto faz uso avançado de visão computacional, sensores de movimento, percepção de profundidade e aprendizado da área em torno de si mesmo. O primeiro dispositivo a suportar a plataforma foi o smartphone ZenFone AR, da Asus (ARAÚJO et al., 2016) e (ASUS, 2017).

### **2.3.1 FERRAMENTAS DE RA PARA DISPOSITIVOS MÓVEIS**

Devido a todo o avanço da Realidade Aumentada e a necessidade de facilitar a implementação da mesma, diversas ferramentas têm surgido para suprir essa demanda, tais ferramentas já contam com implementações de técnicas de rastreamento, interação e outras funcionalidades. Nesta Subseção será apresentado algumas das principais ferramentas que são voltadas para dispositivos móveis.

#### **2.3.1.1 ARToolkit**

Considerado o *Software Development Kit* (SDK) pioneiro de RA, o ARToolkit foi desenvolvido em C e apresentado pela primeira vez em 1999 pelo seu autor Hirokazu Kato. Em 2001 o projeto foi lançado como open-source, sendo muito adotado tanto pela indústria quanto pela comunidade acadêmica. A biblioteca acompanhou a evolução do mercado e foi portada para diversas plataformas, incluindo dispositivos móveis, para a web através do HTML 5 e integração com o motor de jogo Unity3D. O ARToolKit ainda serviu para a criação de diversas bibliotecas derivadas do mesmo, dando suporte assim ao uso do ARToolKit em várias linguagens de programação (DAQRI, 2017) e (BRAGA, 2016).

Fachinetto (2013) afirma que o foco do ARToolKit está no rastreo rápido e cálculo preciso da posição da câmera e de seus marcadores de referência, ou seja, a ferramenta é focada no rastreo através de um marcador quadrado e planar. Basicamente o funcionamento consiste em binarizar (transformar para preto e branco) cada quadro capturado do vídeo e em seguida essa imagem é analisada a fim de identificar esses marcadores. Ao encontrar um marcador a aplicação compara com a imagem previamente cadastrada desse marcador. Caso encontre uma similaridade a biblioteca calcula a orientação e tamanho do marcador a fim de

calcular o posicionamento da câmera, então o marcador é sobreposto por um objeto virtual. Esse rastreamento é tão rápido e preciso que ao manipularmos fisicamente o marcador (girar, aproximar, afastar) o objeto virtual acompanha esse movimento praticamente sem atrasos.

#### **2.3.1.2 Vuforia**

Criado em 2011 pela empresa Qualcomm, o Vuforia é uma das ferramentas mais conhecidas e avançadas para RA, com mais de trezentos mil desenvolvedores. Em 2015 o Vuforia foi adquirido pela Parametric Technology Company (PTC), como plano da empresa em ser líder em prover tecnologia e soluções que misturam os mundos digitais e físicos. O Vuforia ao contrário do ARToolKit é totalmente focado em dispositivos móveis, com suporte a IOS, Android e um plugin para Unity3D. A ferramenta se destaca pelas avançadas técnicas de rastreamento e renderização, proporcionando uma renderização muito mais realista do que seus concorrentes na época que foi lançada (PEDDIE, 2017).

Entre as vantagens do Vuforia vale citar a possibilidade de usar qualquer imagem ou parte desta como marcador, até mesmo uma fotografia. Além disso é possível definir um marcador em tempo de execução e redução de ruído para impedir que o objeto virtual fique tremendo. O sistema ainda é capaz de identificar marcadores mesmo que estes estejam parcialmente cobertos ou saírem parcialmente do campo de visão da câmera, permitindo assim a visualização de um objeto virtual maior do que ultrapassa os limites de tamanho da imagem capturada. Além disso, o Vuforia oferece uma plataforma em Nuvem, onde o desenvolvedor pode cadastrar novos marcadores e objetos virtuais para sua respectiva aplicação (FACHINETTO, 2013).

#### **2.3.1.3 Kudan**

O Kudan é uma empresa Britânica-Japonesa que desenvolve soluções em visão computacional, a empresa disponibiliza um SDK tanto para RA quanto para visão computacional. Seu kit de desenvolvimento para RA é voltado especificamente para dispositivos móveis e Internet das coisas (IoT) podendo executar em plataformas com baixo poder de processamento. Entre as funcionalidades

disponibilizadas temos reconhecimento 2D/3D, rastreamento baseado em marcador e sem marcador utilizando SLAM Visual, entre outros (KUDAN, 2017). Por ter foco em dispositivos com baixo poder de processamento e sua técnica de rastreio principal ser a SLAM optou-se por utilizar essa ferramenta no presente trabalho.

#### **2.3.1.4 Wikitude**

Wikitude é um SDK criado pela empresa alemã de mesmo nome, entre suas funcionalidades temos reconhecimento de imagem e rastreio, suporte a renderização de modelos 3D com sobreposição em vídeo e RA baseada em geolocalização. O Wikitude SDK é uma solução comercial e está disponível nas plataformas IOS e Android, além de um plugin para integração com a engine Unity3D. Além disso a empresa disponibiliza o Wikitude Studio que facilita o processo de desenvolvimento, permitindo a criação do aplicativo arrastando e soltando objetos na tela e sem a necessidade de saber programação (AMIN; GOVILKAR, 2015).

Essa Subseção apresenta apenas algumas ferramentas disponíveis de RA, principalmente aquelas que são mais populares e amplamente utilizadas na indústria. Nos trabalhos de Fachinetto (2013) e de Amin e Govilkar (2015) podem ser encontradas informações a respeito de outras ferramentas de RA em diferentes plataformas.

### **2.4 MODELAGEM 3D**

Muito foi falado sobre os objetos virtuais que são sobrepostos ao ambiente real do usuário, de fato a realidade pode ser aumentada utilizando textos, gráficos e outros objetos de hipermídia (NETTO; MACHADO; OLIVEIRA, 2002). Todavia o maior uso de RA está associada com sobreposição de objetos virtuais tridimensionais (3D). Nesta Subseção são definidos alguns conceitos e fundamentos dessa área da computação.

Um modelo geométrico em sua estrutura mais básica consiste na definição de pontos que forma essa estrutura. Estes pontos são chamados vértices e são

formados pelo conjunto de ternas  $(x,y,z)$ . Esses vértices funcionam como uma ancoragem para outras estruturas geométricas, como segmentos de retas chamados arestas que ligam esses vértices. A partir dessas relações topológicas nosso objeto começa a tomar forma. Ao utilizar o conjunto de arestas e vértices, pode-se criar superfícies chamadas de face (FERREIRA; PELLEGRINO, 2006).

Manssour e Cohen (2006) afirma que existem várias técnicas para representar objetos 3D, todavia a forma mais comum é através de uma malha de polígonos, que é justamente a técnica descrita anteriormente, isto é, define-se um conjunto de vértices no espaço e a ligação desses vértices formam as faces, sendo que essas faces podem ser triângulos ou quadrados. O armazenamento dessa estrutura geralmente é feito através do uso de vetores, matrizes e listas, como o exemplo da Figura 18.

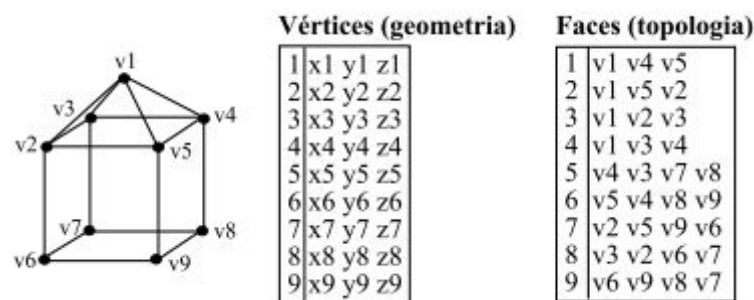


Figura 18 - Objeto tridimensional representado por meio de malha de polígonos.  
Fonte: Manssour (2006)

Segundo Ferreira e Pellegrino (2006), uma das maneiras mais simples e rápida de iniciar a criação de um objeto é por meio de linhas que delimitam seu interior, ficando para o observador reconhecer de maneira intuitiva esse objeto como um sólido. Essa representação é chamada de *wireframe* ou fio de arame. O modelo *wireframe* permite desenvolver estruturas complexas através da modelagem. A partir disso a modelagem consiste em organizar todos esses vértices e arestas de maneira que o objeto comece a tomar forma. A Figura 18 é um exemplo de *wireframe*.

Ao projetar o modelo 3D para uma tela, é realizada uma projeção do que seria esse modelo em uma visão 2D, nesse momento muitas faces não são visualizadas

visto que os objetos são opacos, principalmente as chamadas faces traseiras, que são as faces opostas àquelas que estão voltadas para o observador. Dessa maneira surge a necessidade de eliminar essas superfícies ocultas a fim de facilitar a visão do observador. Um dos algoritmos mais simples para remoção dessas superfícies é o z-buffer (MANSSOUR, 2006) e (FERREIRA; PELLEGRINO, 2006).

Note que nesse momento só temos informações geométricas puras, essas informações só nos fornecem linhas e pontos. Para adicionar realismo aos objetos são necessárias outras informações como cores, rugosidade, tipo do material e como este reage a luz, entre outros. Para adicionar realismo aos objetos é necessário aplicar uma textura nestes. A textura nada mais é que uma imagem 2D que é aplicada na malha poligonal. Para realizar a correta aplicação da textura na malha é necessária fazer um mapeamento de textura, o qual vai associar cada parte da imagem com os respectivos polígonos do modelo (MANSSOUR, 2006).

Observe a Figura 19, nela temos uma estrutura de dados para representar um cubo com seus respectivos vértices, arestas e faces.

<pre>typedef float ponto3[3]; typedef int  aresta[2]; typedef int  face[4];  ponto3 vertices[8] = /* vertice */ {{-1.0, -1.0, -1.0}, /* 0 */  {-1.0, -1.0,  1.0}, /* 1 */  {-1.0,  1.0, -1.0}, /* 2 */  {-1.0,  1.0,  1.0}, /* 3 */  { 1.0, -1.0, -1.0}, /* 4 */  { 1.0, -1.0,  1.0}, /* 5 */  { 1.0,  1.0, -1.0}, /* 6 */  { 1.0,  1.0,  1.0}} /* 7 */</pre>	<pre>aresta arestas[24] = {{0,1},{1,3},{3,2},{2,0},  {0,4},{1,5},{3,7},{2,6},  {4,5},{5,7},{7,6},{6,4},  {1,0},{3,1},{2,3},{0,2},  {4,0},{5,1},{7,3},{6,2},  {5,4},{7,5},{6,7},{4,6}};  face cubo[6] = {{ 0, 1, 2, 3},{ 5, 9, 18, 13},  {14, 6, 10, 19},{ 7, 11, 16, 15},  { 4, 8, 17, 12},{22, 21, 20, 23}};</pre>
---	---

Figura 19 - Estruturas de dados de um cubo tridimensional.  
Fonte: Ferreira e Pellegrino (2006).

Note que na Figura 19 só temos dados a respeito da estrutura base de apenas um cubo, não tem informações de textura e nem outros atributos do objeto 3D. É possível criarmos manualmente essas informações utilizando uma linguagem de programação, todavia para um objeto complexo torna-se inviável fazer manualmente dessa forma. Imagine um cenário completo com milhares de objetos, cada objeto formado por milhares de vértices, arestas e faces. Para solucionar tal



problema foram criadas ferramentas para ajudarem no processo de modelagem 3D.

Segundo Gonçalves (2016) a modelagem poligonal é uma técnica de modelagem manual. Na modelagem manual o artista tem total controle sobre a construção do modelo usando muitas ferramentas que fazem analogia com o processo de escultura tradicional. Entre os softwares de modelagem 3D vale destacar o Blender, software livre profissional de modelagem (BLENDER, 2017). O Autodesk Maya que é muito utilizado na indústria do cinema, jogos e arquitetura (AUTODESK, 2017). E o Autodesk 3D Studio Max que possui muitos recursos iguais ao Maya, todavia está disponível apenas para o windows e tem sido de grande uso na indústria de jogos (AUTODESK, 2017).

### 3 METODOLOGIA

Este trabalho aborda a aplicação da Realidade Aumentada do tipo visualização indireta (visão por vídeo baseado em monitor) utilizando um dispositivo móvel. A técnica de rastreo utilizada foi a *Simultaneous Mapping and Localization* (SLAM), pois é uma técnica não intrusiva em que não é preciso nenhum conhecimento prévio da cena ou marcador fiducial (SIMÕES, 2011; SAÇASHIMA, 2011). A aplicação desenvolvida é capaz de efetuar o download de objetos tridimensionais de uma loja virtual (criada para essa aplicação) e aumentar a realidade do usuário com esses objetos, permitindo uma melhor visualização do produto em seu ambiente. Com intuito de facilitar o entendimento do leitor, o escopo proposto está apresentado como diagrama na Figura 20.

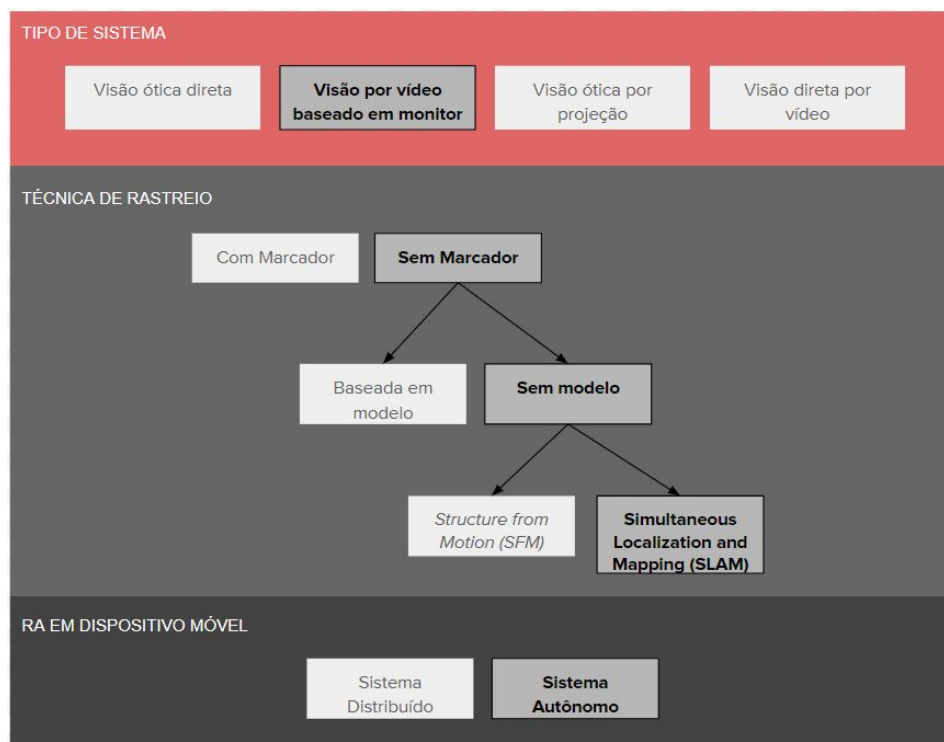


Figura 20 - Classificação da aplicação desenvolvida.

Todos os artefatos necessários para atingir os objetivos, como o código da aplicação, da loja virtual e os modelos 3D, foram disponibilizados publicamente no GitHub, no website: <https://github.com/secato/tcc>.

O testes foram realizados em um smartphone com processador Exynos 1.9GHz octa core, 3GB de memória RAM e sistema operacional Android 6.0 Marshmallow.

Basicamente o desenvolvimento consiste em três etapas, sendo: o desenvolvimento da loja virtual; a obtenção de modelos tridimensionais de alguns produtos para disponibilizar nessa loja; e a aplicação móvel de RA. Nas próximas seções, segue uma descrição de cada etapa.

### 3.1 DESENVOLVIMENTO DA LOJA VIRTUAL

Com a utilização do *framework* Bootstrap, foi criado um site que simula o visual de uma loja virtual, apresentando alguns produtos com suas respectivas características, imagens e modelos 3D disponíveis para a aplicação realizar o download. Como o foco do presente trabalho não é implementar uma loja virtual completa, foram utilizadas apenas algumas páginas estáticas para apresentar os modelos que serão descritos na Subseção 3.2.

Com a finalidade de visualizar a página antes mesmo de codificar, foram criados protótipos de baixa fidelidade para ter uma idéia principalmente sobre a diagramação do site e disposição dos elementos visuais. O protótipo representado na Figura 21 apresenta um esboço da página principal, onde há um cabeçalho com barra de pesquisa e categorias, que é padrão para todas as outras páginas, no centro da página os produtos em destaque e no final da página um rodapé

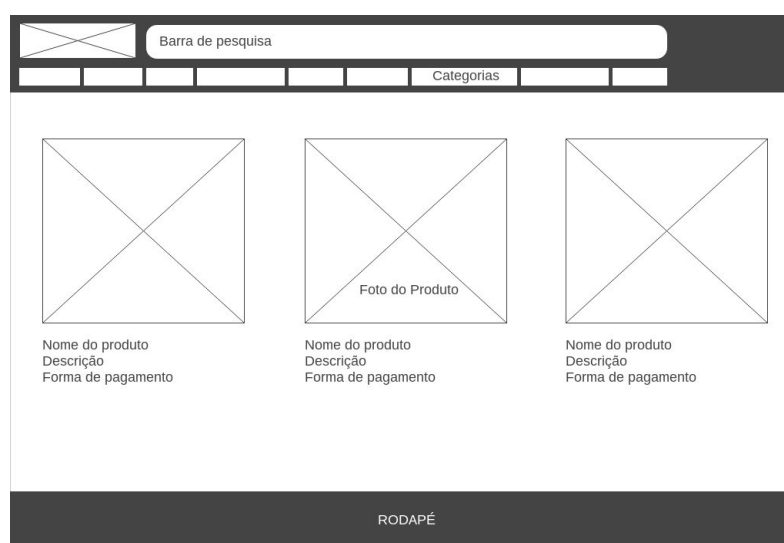


Figura 21 - Protótipo da página principal da loja.

Na Figura 22 tem-se o protótipo criado para uma página de detalhes do produto, em que para cada produto tem uma página com uma visão mais detalhada de suas

características, assim como outras imagens. Também foi adicionado um link de download da aplicação de RA desenvolvida neste trabalho.

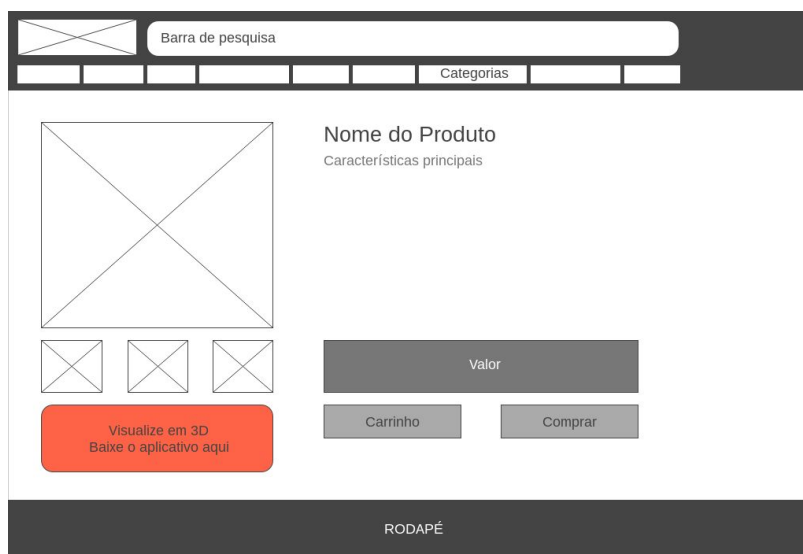


Figura 22 - Protótipo da página de visualização de um produto.

É importante notar que o protótipo já foi idealizado de acordo com os componentes disponíveis no *framework* Bootstrap. Por exemplo, a barra superior pode ser feita utilizando o componente *navbar* do *framework*, o mesmo vale para os botões e até mesmo a diagramação da página. Assim, o desenvolvimento do site foi realizado utilizando o *framework front-end* Bootstrap (BOOTSTRAP, 2017), visto que este já atende diversas necessidades de uma aplicação web moderna, como interface amigável e padronizada através dos componentes prontos que o framework traz. Além disso, o *framework* possibilita o site ter o atributo de responsividade, isto é, a capacidade da página se adaptar a diversos tamanhos de telas, como por exemplo, a de smartphones.

Para escrever o código da loja virtual, recomenda-se o uso de um editor de texto voltado para desenvolvimento web a fim de agilizar o trabalho e evitar erros. Neste projeto foi utilizado o Visual Studio Code, por ser um software gratuito e otimizado para desenvolvimento de aplicações web, já contando com recursos como autocompletar e depuração de JavaScript nativo (MICROSOFT, 2017).

Na Figura 23 é possível visualizar as páginas após codificadas e interpretadas no navegador. Na Figura 23A temos a página principal e em 23B a página de detalhes do produto impressora.

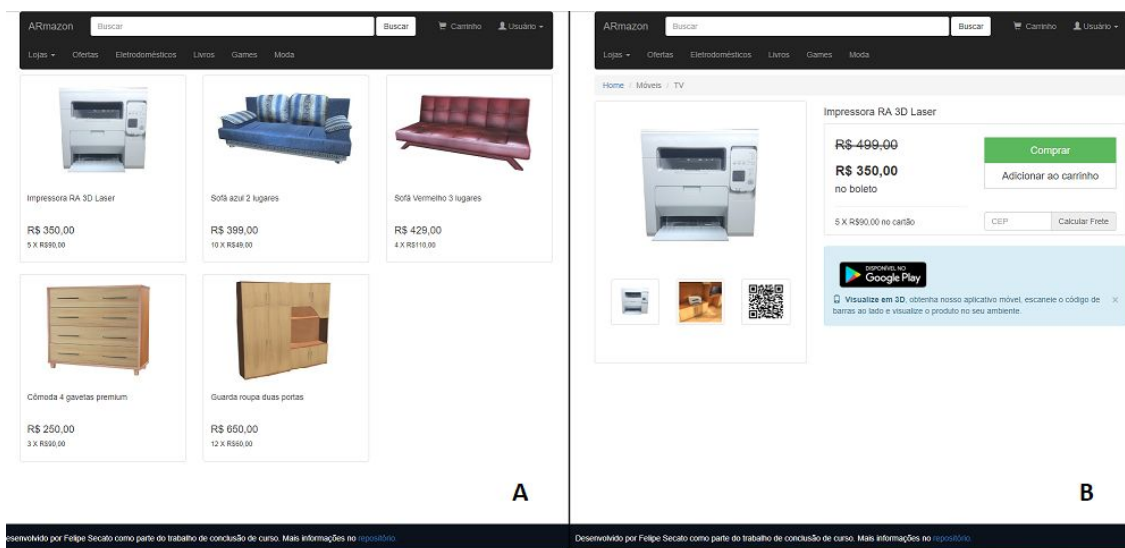


Figura 23 - Visual da loja virtual: (A) página inicial da loja; (B) página de detalhes do produto cômada.

A Figura 24 apresenta a loja sendo acessada de um dispositivo móvel, em que é aplicado o atributo de responsividade, isto é, a capacidade de se adaptar a telas de diferente tamanho.

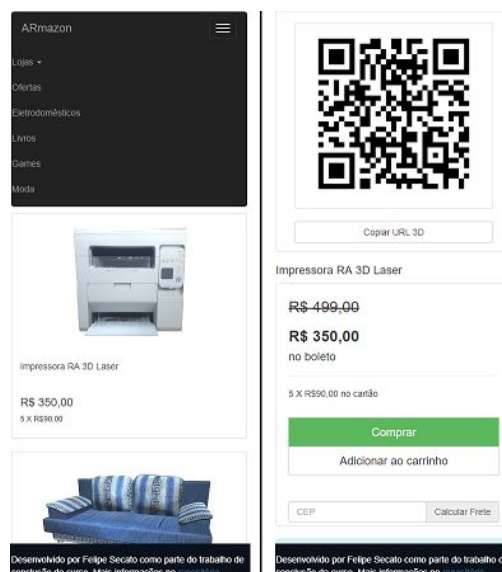


Figura 24 - Visual da loja virtual quando acessado de um dispositivo móvel.

Como o objetivo da loja é servir como apoio para simular o ambiente de uso da aplicação móvel, não foi adicionado nenhum comportamento aos botões, campos de busca, entre outros, simulando somente a aparência de uma loja real. Para o caso da loja virtual ser acessada diretamente pelo smartphone, foi colocado um botão (disponível somente para acesso de um dispositivo móvel) que permite que o usuário copie a URL do modelo 3D ao invés de escanear o QR Code.

Com o site construído e testado é necessário disponibilizá-lo online. Portanto, o site foi disponibilizado em <<https://secato.github.io/tcc/loja/index.html>>.

### 3.2 AQUISIÇÃO DOS MODELOS 3D DOS PRODUTOS

Existem diversas maneiras de se obter modelos 3D de diferentes produtos na Internet, podendo estes serem pagos ou gratuitos e em diferentes formatos de arquivo. O Unity aceita importar modelos 3D a partir de extensões conhecidas, como: “.fbx”, “.dae” (Collada), “.3ds”, “.dxf”, “.obj” e “.sk” (UNITY, 2017).

Entretanto, o Unity não aceita, por padrão, a importação em tempo de execução de objetos nos formatos já citados anteriormente. Apesar de existirem *scripts* na loja do Unity para realizar tal tarefa, não foi identificado nenhum que realize a função de importar no Android os formatos mais populares como “.obj” e “.dae” em tempo de execução. Diante disso, foi necessário utilizar um recurso do Unity chamado *AssetBundle*, que é um arquivo contendo objetos específicos para determinada plataforma, como modelos, texturas, áudios e até mesmo uma cena completa, e que podem ser carregados em tempo de execução. A documentação do Unity traz detalhadamente como realizar o uso de *AssetBundles* (UNITY, 2017).

Como a tarefa de modelagem 3D não faz parte do escopo do presente trabalho e devido a limitação citada anteriormente, optou-se por obter os modelos 3D dos produtos diretamente da loja integrada do Unity. A vantagem de se obter diretamente da loja é que o objeto 3D já vem pronto para uso, sendo necessário pouca ou nenhuma configuração, diferentemente quando se importa os modelos de outras aplicações, em que é necessário configurar os parâmetros de importação e criar os

materiais (componentes gráficos) que são responsáveis por trazer realismo ao objeto.

O pacote de modelos 3D utilizado neste trabalho e exibido na Figura 25, pode ser encontrando na *Asset Store* pelo seu respectivo nome: *Pack Gesta Furniture #1*. Este pacote, além de gratuito, traz diversos modelos 3D já modelados com uma técnica chamada *low poly*, onde se utiliza o mínimo de polígonos em conjunto com outras técnicas para representar um objeto com resultados similares a um modelo *high poly* (alta densidade de polígonos) mas que requer menor poder de processamento.

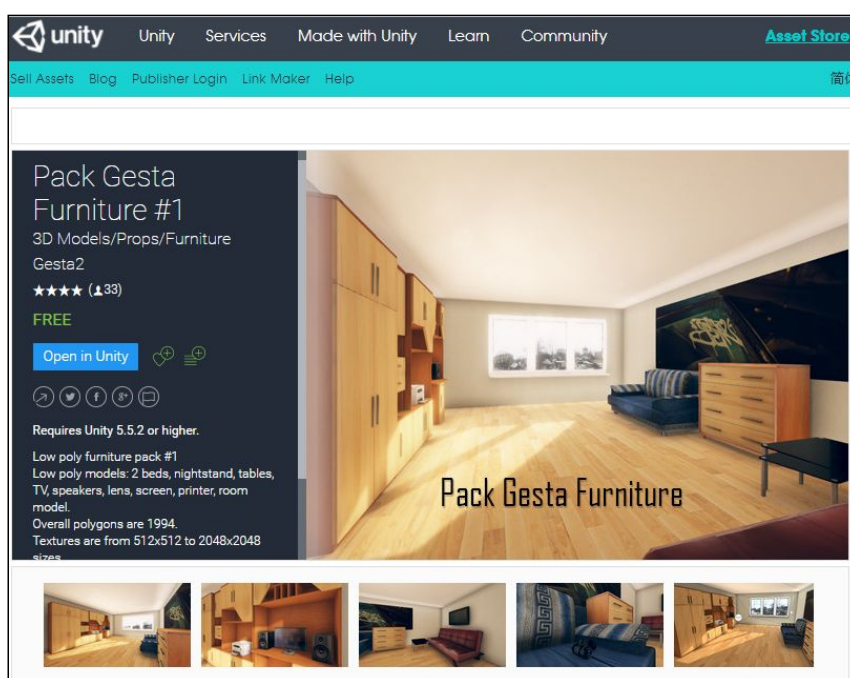


Figura 25 - Pacote de modelos 3D utilizado neste trabalho.

### 3.3 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL

Para desenvolvimento da aplicação móvel foi utilizada a *engine* Unity versão 5.5.4p4 (UNITY, 2017) em conjunto com a biblioteca de RA Kudan na versão 1.5 (KUDAN, 2017). O foco do Kudan é rastreo baseado em características naturais (isto é, rastreamento sem marcador), mais especificamente o SLAM, atendendo assim às necessidades do projeto. Além disso a biblioteca foi construída com foco em dispositivos móveis e Internet das Coisas, logo seus algoritmos são otimizados para

dispositivos com pouco poder de processamento. O Kudan está disponível para Android, IOS e como um plugin para a *engine* Unity.

A vantagem de utilizar a Unity como plataforma de desenvolvimento é sua capacidade de portabilidade, em que um único código fonte pode ser compilado para diversas plataformas. Além disso, por se tratar de uma ferramenta para desenvolvimento de jogos, ela possui todos os recursos necessários para criar a interface e realizar as manipulações com os objetos 3D, como escala, rotação, translação.

Como o projeto em questão tem como plataforma de destino o sistema operacional Android, é necessário realizar o download do Android SDK e do Java JDK e configurar no Unity para que este possa compilar para a plataforma em questão (Unity, 2017). Nesse passo é importante notar que o plugin de RA Kudan requer a versão mínima do android 4.0.3 para seu correto funcionamento (Kudan, 2017).

Para auxiliar a escrever o scripts da aplicação foi utilizada a IDE MonoDevelop a qual já vem na instalação padrão do Unity. A Figura 26 traz a aba *Project* do Unity na qual é possível visualizar a organização dos arquivos da aplicação.

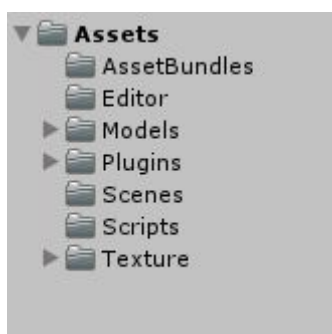


Figura 26 - Organização dos arquivos da aplicação.

A pasta AssetBundles é a pasta de destino após os modelos 3D serem empacotados como Asset Bundle. A pasta editor contém um script que adiciona a funcionalidade de criar Asset Bundle ao editor do Unity. A pasta models contém os modelos 3D com suas respectivas texturas e materiais. A pasta plugins contém os plugins de terceiros utilizados no projeto, como o Kudan e o leitor de código de barras. A pasta scenes contém as cenas “DownloadModel” e “RA”, na pasta scripts estão os códigos



utilizados na aplicação e em Texture estão as imagens que foram utilizadas para a interface, tela inicial, entre outros.

Um conceito importante no Unity é o conceito de Cena. Uma cena no Unity é como se fosse um nível, ou uma fase fazendo uma analogia com um jogo, cada cena é independente e contém seu próprio conjunto de objetos (Unity, 2017). A arquitetura da aplicação desenvolvida para este trabalho assim como as cenas que a compõem podem ser visualizadas na Figura 27, em que pode-se ver a aba de hierarquia do Unity da cena de nome DownloadModel (Figura 27A), contendo os objetos que fazem parte da mesma. O mesmo se aplica à Figura 27B, porém para a cena de nome RA. Já na Figura 27C temos uma representação em UML<sup>5</sup> de uma classe estática utilizada como apoio no projeto. Como as cenas no Unity são independentes e tudo que está em uma cena se perde ao mudar para outra, foi necessário criar essa classe estática para manter os modelos 3D obtidos da Internet pela cena DownloadModel e disponibilizá-lo para a cena RA, visto que uma classe estática no Unity não precisa estar associada à nenhum objeto e é visível por toda a aplicação.

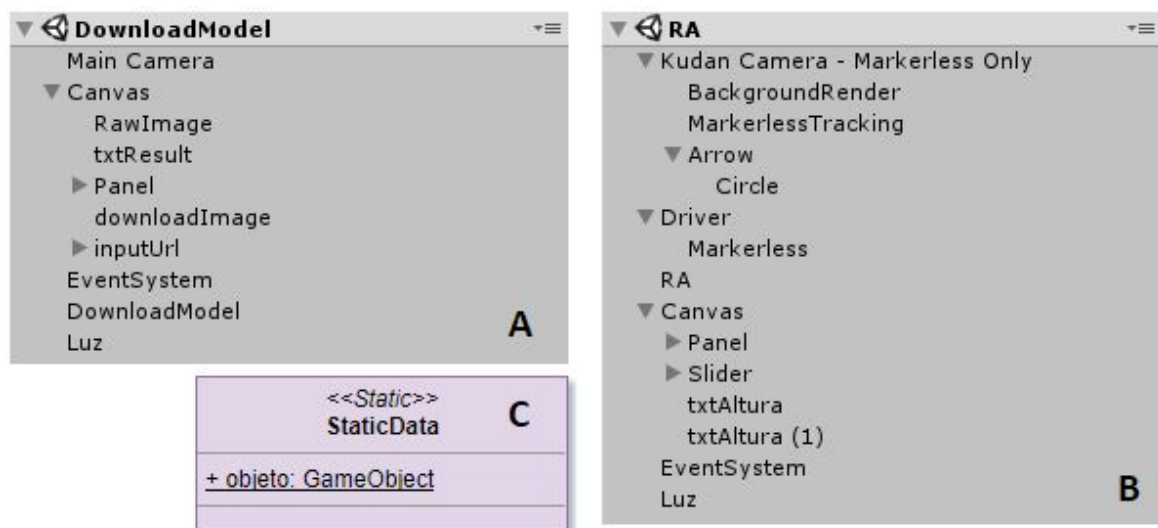


Figura 27 - Elementos da aplicação no Unity, em (A) temos os objetos que fazem parte da cena DownloadModel. Em (B) os objetos que fazem parte da cena RA e em (C) a representação da classe estática StaticData.

A seguir são detalhados os componentes de cada cena bem como seu respectivo funcionamento.

<sup>5</sup> UML - Unified Modeling Language (Linguagem de Modelagem Unificada)

### 3.3.1 Cena DownloadModel

A cena DownloadModel é a cena inicial da aplicação, sendo esta a primeira tela que o usuário terá acesso. Essa cena é responsável por trazer todos os recursos para obter os modelos 3D da Internet. Além disso, ela funciona como uma pré-visualização do objeto 3D em questão antes dele ser visualizado através da Realidade Aumentada. A cena fornece um leitor de código de barras QR Code e alternativamente um meio de obter um modelo 3D passando a URL do mesmo. A respeito dos objetos que compõe a cena, tem-se: uma câmera principal, que é um objeto do Unity o qual vai renderizar a parte gráfica da aplicação; o objeto canvas, que traz a interface gráfica do usuário e que contém diversos botões, campo texto e outros; e o objeto Luz, que é uma *Directional Light*, um objeto padrão da Unity para criar uma luz artificial dando um aspecto mais realista a pré-visualização do objeto.

O objeto responsável por trazer a funcionalidade principal dessa cena é o objeto DownloadModel, sendo capaz de realizar o download dos modelos 3D, copiá-los para a classe estática StaticData e instanciar o objeto para pré visualização. O código relativo a função de download está disponível no Apêndice I. Na Figura 28 é possível visualizar a cena pelo editor do Unity, assim como a configuração do objeto DownloadModel com os parâmetros configurados no script.

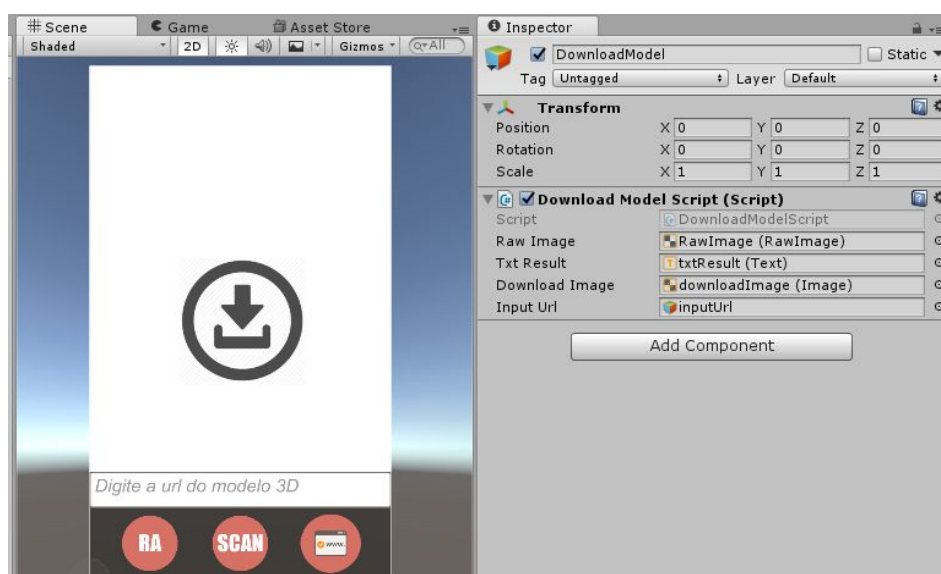


Figura 28 - Cena DownloadModel visualizada pelo editor do Unity e configuração do objeto DownloadModel.

Já a Figura 29 traz uma demonstração da cena 1 DownloadModel em funcionamento. Ao clicar no botão SCAN, a câmera é iniciada como leitor de QR Code, conforme Figura 29A. Ao escanear o QR Code, é obtido a URL do modelo 3D, que após o término do download, fica disponível para pré-visualização conforme Figura 29B. Neste momento, se quiser, o usuário pode rotacionar, aumentar e diminuir a escala do objeto.



Figura 29 - Demonstração da cena DownloadModel, em (A) Escaneando qr code da loja virtual; (B) Pré-visualizando o produto.

### 3.3.2 Cena RA

A cena RA é responsável pela Realidade Aumentada da aplicação. Esta cena é composta por três objetos principais: o Kudan Camera, que faz acesso à câmera do dispositivo e que possui os mecanismos necessários para efetuar a marcação ao posicionar e renderizar o modelo 3D; o objeto Driver, que possui os mecanismos necessários para rastrear e exibir o modelo 3D; e o Canvas, que contém a interface do usuário e que nesse caso conta com uma funcionalidade importante que é uma barra lateral deslizante, em que pode-se definir a altura do chão artificial que o Kudan utiliza para posicionar e realizar uma escala proporcional no objeto,

forneendo assim um meio para definir a distância entre o dispositivo móvel e a superfície que o objeto será projetado.

O objeto RA é responsável pela lógica principal da cena, isto é, aumentar a realidade com o modelo 3D obtido previamente utilizando as execuções da API do plugin Kudan. As funcionalidades principais do script podem ser visualizadas no Apêndice II. Na Figura 30 temos a cena vista pelo editor do Unity, bem como o objeto RA e seus parâmetros no script.

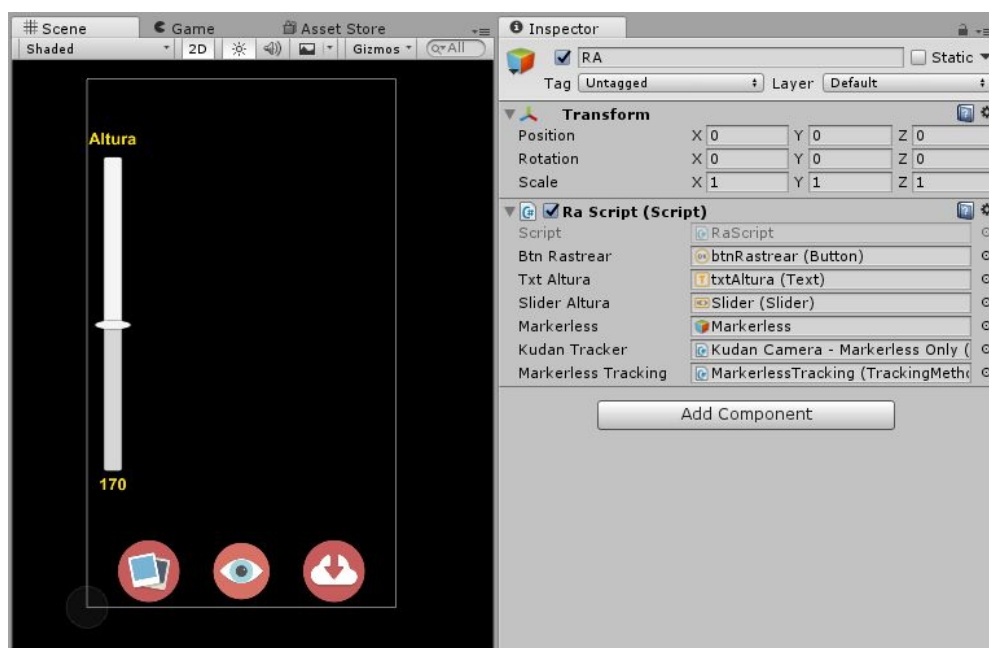


Figura 30 - Cena RA visualizada pelo editor do Unity e configuração do objeto RA.

Na Figura 31 e 32 é possível ver a cena RA em funcionamento. Antes de posicionar o objeto temos um indicador vermelho que mostra onde o modelo 3D será posicionado após o clique no botão de rastreo (representado pelo ícone em formato de olho).

Um aspecto importante dessa cena é a barra lateral deslizante que é utilizada para configurar o valor de altura entre o dispositivo móvel e a superfície que o modelo 3D será projetado, isso ajuda a localizar o objeto de maneira mais eficiente na hora do rastreo e a realizar uma escala proporcional no tamanho do objeto. O Kudan não faz essa escala automaticamente pois é complexo detectar a distância entre o dispositivo móvel e a superfície que o objeto irá ser projetado utilizando apenas uma

câmera (SLAM Monocular). Na Figura 31A nota-se que o indicador de onde o objeto será posicionado é bem maior do que na Figura 31B, pois a altura na Figura 31A está configurada para 50 centímetros ao passo que na Figura 31B a altura está configurada para 1,70m.

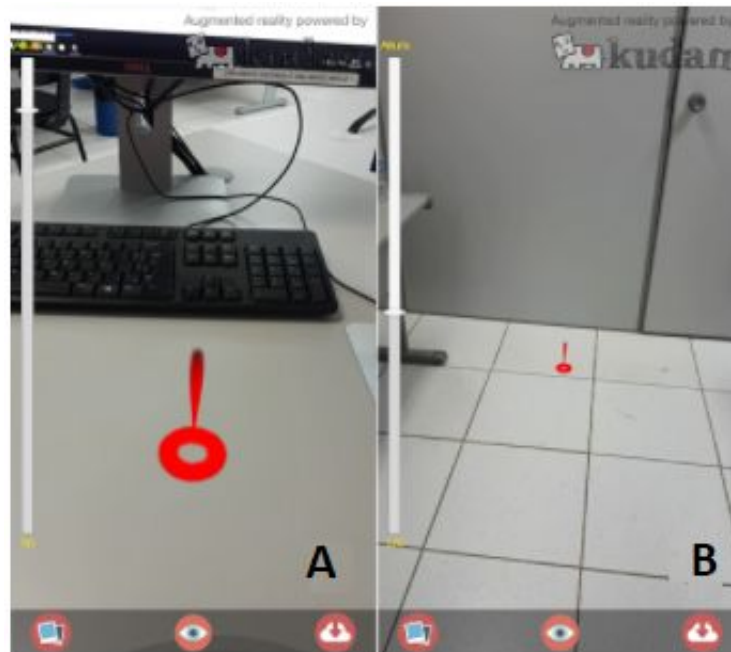


Figura 31 - Demonstração da cena RA: Indicador de posicionamento com altura configurada em 50 centímetros em (A) e altura configurada em 1,70m em (B).

Na Figura 32, o rastreo foi iniciado e o modelo 3D é posicionado onde estava marcado vermelho anteriormente. À partir desse momento, o usuário pode se movimentar ao redor da cena que o objeto continuará posicionado no mesmo local. É recomendado que se faça movimentos suaves e segure o dispositivo móvel firmemente para evitar trepidação no rastreo.



Figura 32 - Demonstração da cena RA: Rastreio do objeto impressora (A) e do objeto cômoda(B).

### 3.4 DISPONIBILIZAÇÃO DA APLICAÇÃO NA PLAY STORE

O aplicativo foi disponibilizado na Play Store, que é a loja oficial do Android. Isso facilitou a distribuição do aplicativo e permitiu maior comodidade dos usuários para instalar o mesmo na hora de realizar os testes. Para a disponibilização do aplicativo, foi necessária uma conta de desenvolvedor da Play Store, que permite acesso à central de publicação, podendo informar o nome do aplicativo que está sendo disponibilizado, sua descrição, suas imagens e outras informações. O aplicativo ficou disponível em: <<https://play.google.com/store/apps/details?id=com.Secato.ArView>>.

No Unity, foi necessário realizar poucas configurações. Basicamente, define-se o ícone do aplicativo, a tela de carregamento (nesse caso é exibida o brasão da UFES). As configurações mais importantes ficam a cargo de definir a API mínima do aplicativo, isto é, o requisito mínimo de plataforma que é o Android 4.0.3 (devido a requisitos do Kudan, conforme abordado na Subseção 3.3), e o Bundle Identifier, que é um identificador único para o aplicativo, para que este seja identificado na Play Store.

### 3.5 UTILIZAÇÃO DO APLICATIVO E AVALIAÇÃO DOS USUÁRIOS

Com todas as etapas anteriores finalizadas, o usuário tem disponível o ambiente necessário para testar a aplicação, a Figura 33 traz um fluxograma de um cenário de uso esperado pela aplicação.

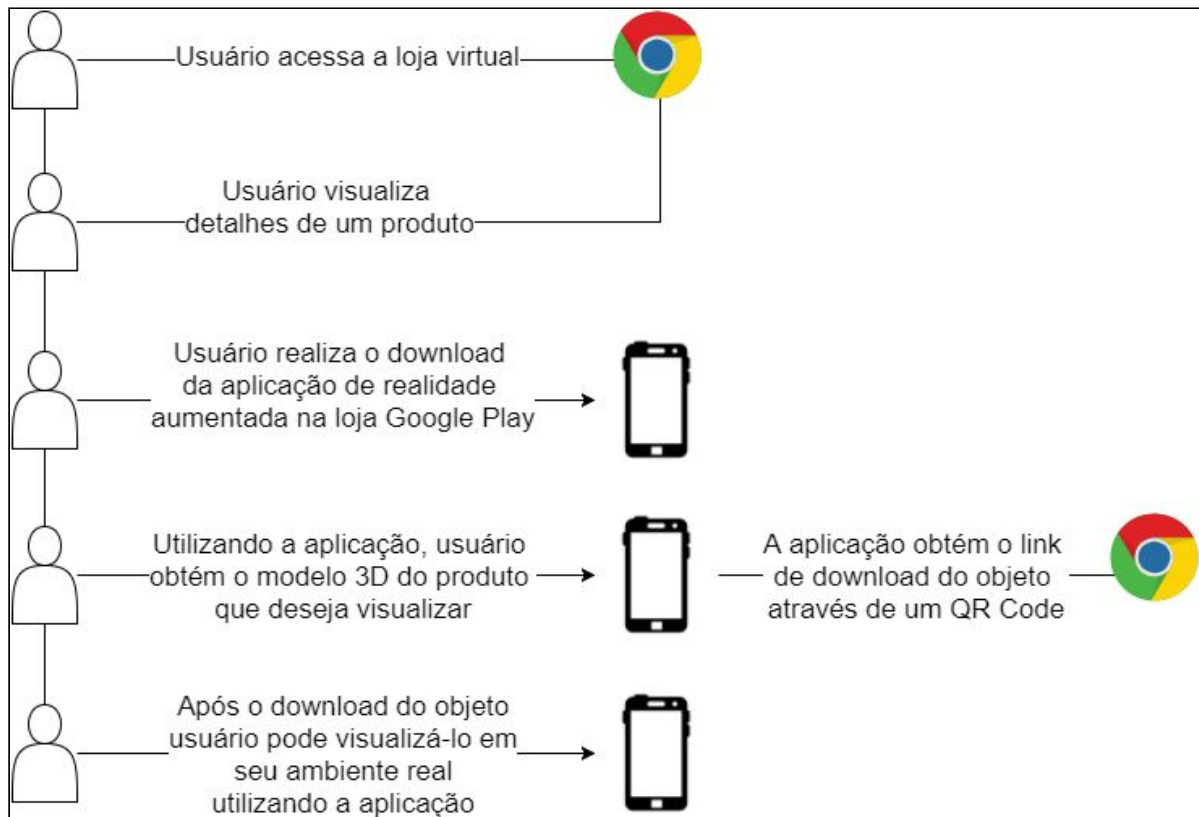


Figura 33 - Fluxo de uso da aplicação.

Sendo assim, foi solicitado para que os usuários testassem a aplicação e avaliassem, por meio de uma pesquisa qualitativa (questionário), se o projeto atingiu o objetivo proposto.

O questionário foi planejado de forma a examinar uma grande amostra de resultados e por isso optou-se pelo uso de questões objetivas, visto que estas podem ser mais facilmente quantificadas e analisadas, além de terem um baixo custo de tempo para serem respondidas (VIALI, 2017). O questionário pode ser conferido no Apêndice III e os resultados deste são discutidos na Subseção 4.

#### 4. ANÁLISE DOS RESULTADOS

Nesta seção, são apresentadas as análises referentes aos questionários respondidos pelo público que realizaram testes com o aplicativo proposto. Os testes foram feitos em ambiente não controlado, sendo divulgado o aplicativo e o questionário disponível no Apêndice III pela Internet através de redes sociais. Assim, os usuários puderam executar o aplicativo em seus próprios dispositivos e em seus respectivos ambientes (loais físicos onde se encontravam). Foram obtidas 37 respostas, a maior parte do público que respondeu o questionário já possuía familiaridade com tecnologia, visto que os testes tiveram maior participação dos usuários pertencentes aos grupos acadêmicos de Sistemas de Informação e de Ciência da Computação da UFES.

Nas questões 1 e 2, relacionadas com o contato do público sobre a Realidade Aumentada (Figura 34 e 35), pode-se perceber que 70% das pessoas já conheciam a RA, mas mesmo assim, somente 54,5% tiveram contato com essa tecnologia. Isso mostra que, mesmo sendo um público com maior contato à tecnologia, a RA ainda não o atinge completamente, confirmando assim sua utilização e/ou publicação restrita, conforme abordada na Subseção 1.1. Diante disso, aplicativos de RA, como o desenvolvido neste trabalho, podem abranger novos mercados e enriquecer a experiência do usuário em diversas áreas.



Figura 34 - Respostas da questão 1 do questionário.





Figura 35 - Respostas da questão 2 do questionário.

Na questão 3 do questionário (Figura 36), relacionada com a dificuldade do público em realizar compras pela Internet, nota-se que 78,4% do público já teve algum tipo de dúvida ao comprar um produto pela Internet, deixando para comprar tal produto em uma loja física. Esse resultado mostra que a maioria dos consumidores têm dificuldades e falta de confiança ao comprar pela Internet, pois o público brasileiro, em sua maioria, está mais habituado a comprar produtos onde sua comparação e visualização física é mais comum, tornando-se muito diferente da simples visualização de fotos de um produto online. Além disso, a variedade de produtos e de opções personalizadas dificulta a tomada de decisão do usuário que, ao confrontar as diversas variáveis como cor, tamanho e ambiente de destino, não consegue chegar a uma conclusão devido à limitação encontradas nas maneiras tradicionais de visualização, conforme abordado na Subseção 1.1.

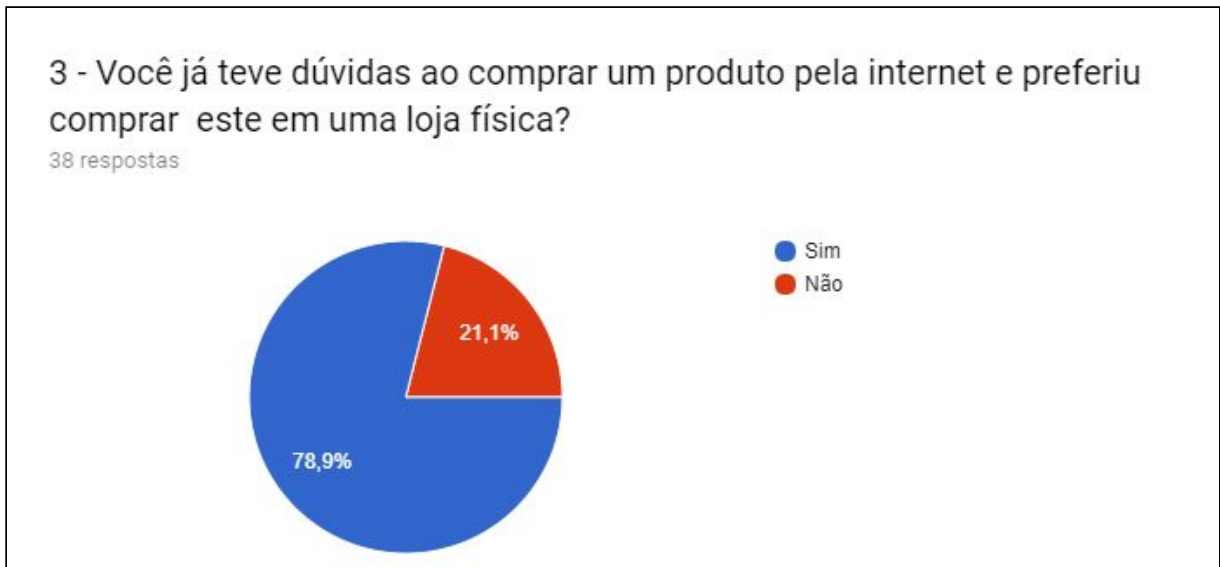


Figura 36 - Respostas da questão 3 do questionário.

A questão 4 (Figura 37) complementa a questão anterior, pois diante do fato da dúvida e da visualização do produto ser limitada, conforme abordada na questão 3 (Figura 36), a maioria do público (75,7%) avaliou que a possibilidade de visualizar o produto em seu ambiente antes mesmo de realizar a compra se mostrou muito útil (nota 4) e 24,3% do público avaliou como útil (nota 3), conforme Figura 37. Assim, todo o público obteve alguma utilidade com a aplicação de RA como um meio para visualizar os produtos online, demonstrado assim que o uso da RA se mostrou um avanço nas maneiras tradicionais de visualização, permitindo que o usuário visualize e interaja com o produto em seu ambiente real antes mesmo de realizar a compra, evidenciando a necessidade da utilização de RA por parte do público no que tange compras pela Internet.

O aplicativo também mostrou a viabilidade da RA em outras áreas diferentes do *marketing* e jogos, sendo esses os responsáveis pela grande maioria das aplicações atuais de RA. Ao proporcionar uma melhor visualização e interação, este trabalho mostrou que a tecnologia de RA pode ser utilizada em diversas áreas diferentes, como o comércio eletrônico.

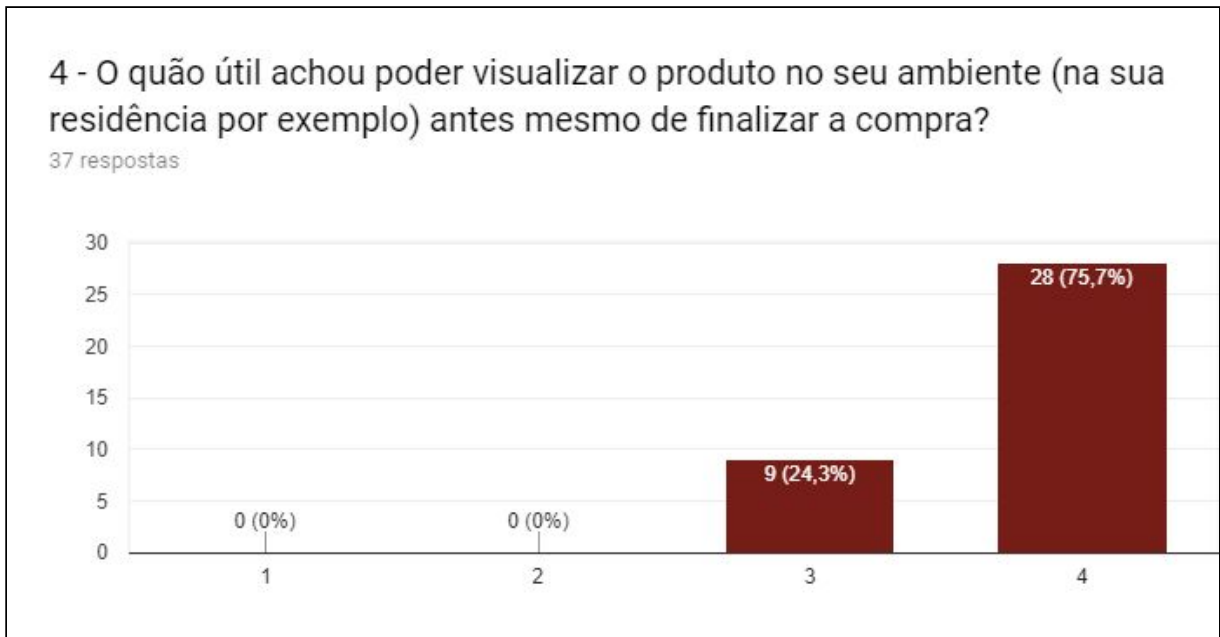


Figura 37 - Respostas da questão 4 do questionário.

A questão 5 também se relaciona com a questão 3, pois diante da dúvida de compra, o aplicativo mostra-se útil para ajudar na tomada de decisão da compra. Conforme as respostas apresentadas na Figura 38, 45,9% do público respondeu que o aplicativo ajudaria muito (nota 4) na tomada de decisão de compra de um produto no qual ainda está em dúvida. Já 24,3% do público respondeu que o aplicativo ajudaria (nota 3) na tomada de decisão, provando que a RA pode ser utilizada como auxílio no processo de compra de produtos pela Internet, enriquecendo assim a visualização do usuário conforme abordado na seção 1.1 deste trabalho. Todavia 21,6% do público respondeu que o aplicativo pouco ajudaria (nota 2) na tomada de decisão e 8,1% respondeu que o aplicativo não ajudaria nada (nota 1). A análise de tais motivos são apresentadas junto à análise da questão 6 do questionário (Figura 39).



Figura 38 - Respostas da questão 5 do questionário.

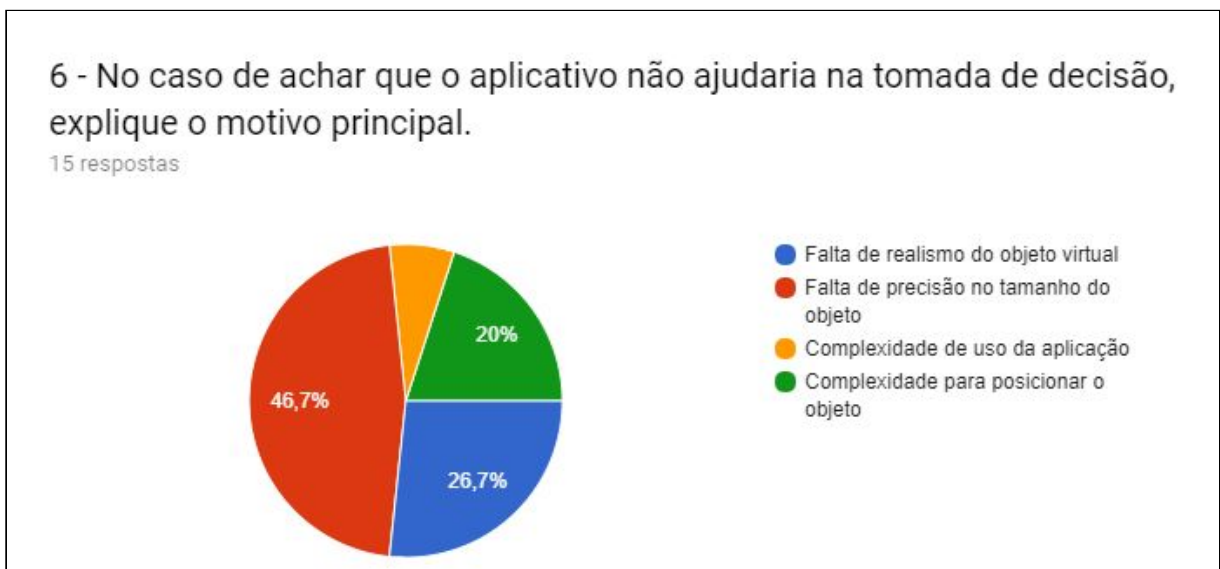


Figura 39 - Respostas da questão 6 do questionário.

O principal motivo (questão 6, Figura 39) apontado por 46,7% do público que respondeu que o aplicativo não ajudaria na tomada de decisão (questão 5, Figura 38), foi a falta de precisão no tamanho do objeto. Acredita-se que esse problema ocorreu pela necessidade de intervenção manual do usuário para informar a altura entre o dispositivo e a superfície que o objeto era projetado, pois só assim que a aplicação poderia realizar uma escala proporcional no tamanho do objeto. Além

disso, a aplicação faz uma estimativa do tamanho do objeto conforme o ângulo que o usuário segura o smartphone, esse comportamento é apresentado na Figura 40.

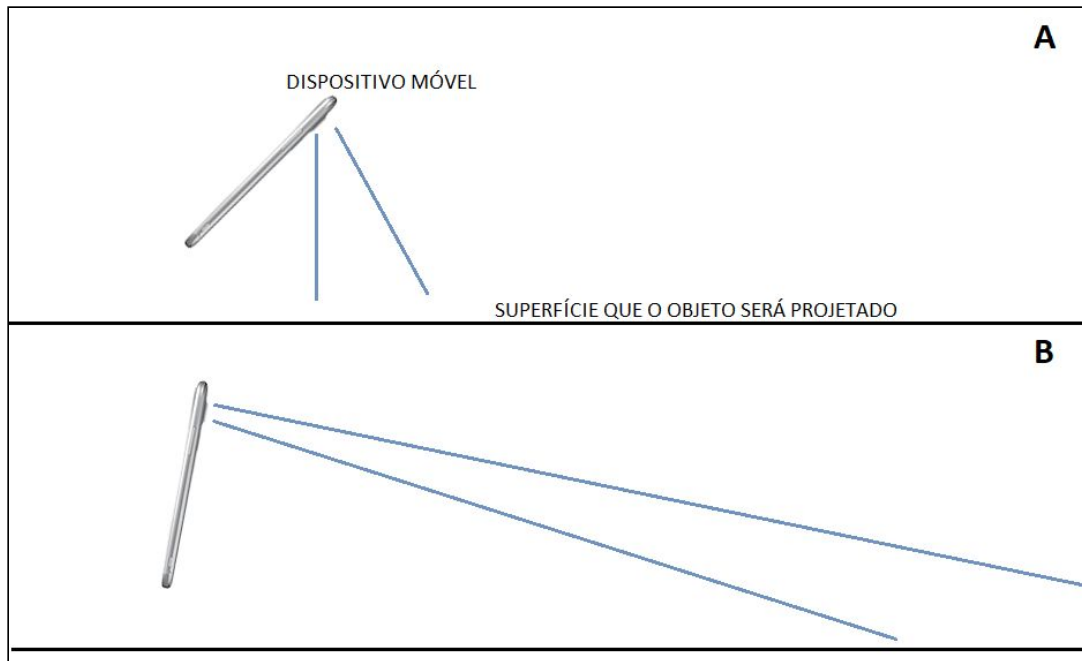


Figura 40 - Lógica de projeção do modelo 3D pela aplicação.

Com o dispositivo móvel menos inclinado, conforme a Figura 40A, a aplicação entende que o usuário está querendo projetar o modelo 3D mais perto dele, realizando uma escala proporcional no tamanho do objeto para que este fique maior quando projetado mais próximo do usuário. Já com o dispositivo mais inclinado, conforme a Figura 40B, a aplicação entende que o objeto será projetado mais distante do usuário e o diminui proporcionalmente. Contudo, isso pode gerar uma falsa proporção no tamanho do objeto, pois a aplicação pode entender que o usuário quer projetar o objeto muito mais distante do que realmente deseja devido a complexidade de se determinar obstáculos, dada a própria natureza da técnica SLAM que faz identificação apenas de marcos visuais. Uma das maneiras de amenizar esse problema seria o usuário também configurar manualmente a profundidade (a distância) que se deseja projetar o produto, assim como é feito com a altura. Todavia, essa é uma maneira manual e que também dá margem para erro caso o usuário coloque este valor erroneamente.

O segundo motivo, apontado por 26,7% do público (Figura 39), é a falta de realismo do objeto virtual, que acontece por três principais causas. A primeira causa é que para um objeto parecer mais realista e realmente inserido em uma cena, ele deve sofrer influência da luz do ambiente, assim como gerar sombra, reflexo, entre outras características. Isso não acontece pois o objeto virtual não sofre nenhum desses efeitos da luz ambiente deixando-o com um aspecto não tão realista, neste caso poderia ser utilizado uma luz artificial para simular esse comportamento. Porém, esta luz deveria ter a mesma coloração (amarelada ou azulada por exemplo), intensidade e posição da luz do ambiente real, sendo necessário intervenção manual para configurar estes parâmetros ou um algoritmo que consiga captar e calcular essas parâmetros a partir da luz ambiente, o que é demasiadamente complexo e de custo computacional alto.

A segunda causa é que objetos complexos e realistas podem acarretar maior peso de processamento na aplicação, podendo inviabilizar o uso da tecnologia em hardwares mais modestos. Já a terceira causa está relacionada ao formato de arquivo dos modelos 3D, pois esses formatos mais populares e públicos, como “.obj” ou “.dae”, armazenam somente a estrutura do objeto 3D (a malha), conforme abordado na Subseção 2.4. As configurações dos materiais ficam em um arquivo separado e armazenam apenas as características mais simples, ou seja, se na modelagem 3D utilizando o software Blender foi criada um material complexo que dê uma aparência de vidro (com transparência, reflexo, refração) para seu modelo, dificilmente será possível exportar esse material criado no Blender, sendo necessário recriar esse material na aplicação que vai renderizar o modelo (seria necessário criar o material “vidro” no Unity, por exemplo). Isso inviabilizaria a questão da interoperabilidade do modelo 3D, pois cada aplicação teria que ter configurações próprias, ou seja, seria necessário um arquivo de modelo 3D para Unity, outro para Android nativo e outros para as diferentes plataformas que utilizariam o modelo.

O terceiro motivo apontado por 20% do público (Figura 39) é a complexidade para posicionar o modelo 3D durante o uso da aplicação. Parte dessa complexidade ocorre devido a dinâmica de posicionamento da aplicação, conforme já explicado na

Figura 40. Além disso, um ambiente muito escuro ou muito homogêneo (um piso totalmente liso e branco, por exemplo) também pode dificultar o posicionamento do objeto, pois a técnica SLAM utilizada neste trabalho detecta marcos visuais como pontos e arestas, conforme abordado na subseção 2.2.3.2. No que tange a questão de posicionamento do objeto, o uso de RA com marcador facilitaria bastante esse processo, com o inconveniente de ter que imprimir e posicionar o marcador manualmente. Já 6,7% do público alegou ter complexidade no uso da aplicação, esse problema poderia ser amenizado com um tutorial guiado na primeira vez de uso da aplicação, como acontece em muitas aplicações móveis atualmente.

Já a questão 7 (Figura 41) relaciona-se à percepção da utilidade da RA para compras online vista pelo dono da loja virtual. Nota-se que 97,3% do público acredita que os usuários iriam sentir-se mais confiantes e confortáveis ao realizar uma compra de um produto online utilizando a RA como apoio na visualização do produto e que isso aumentaria as vendas pela Internet. Essas respostas corroboram com a pesquisa que foi citada na Seção 1, onde o número de vendas pela Internet aumentaria caso a confiança do público aumentasse, principalmente ao comprar produtos de categorias no qual não estão muito habituados.

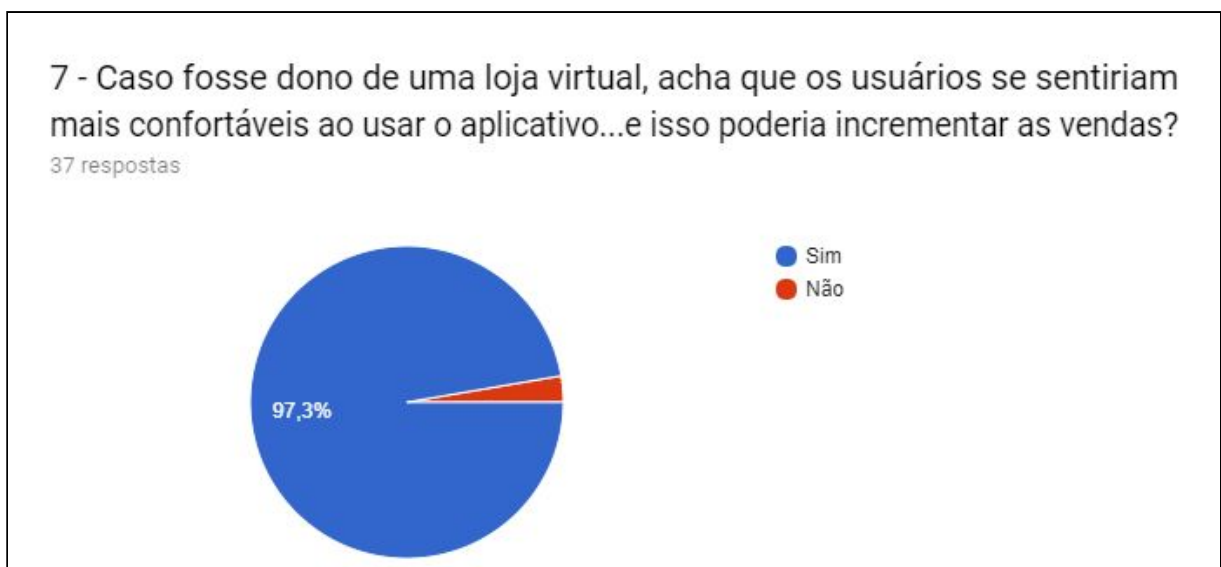


Figura 41 - Respostas da questão 7 do questionário.

Por fim, a questão 8 (Figura 42) avalia os benefícios que a tecnologia da RA forneceria à sociedade como um todo se fosse utilizada por todas as lojas virtuais, tornando-se um padrão. Conforme a Figura 42, 100% do público respondeu que a sociedade se beneficiaria com o uso de tal tecnologia para realizar compras online em lojas virtuais, pois a aplicação possibilita que o usuário tenha maior confiança ao comprar e visualize o produto previamente. A questão 8 mostrou que a aplicação da RA como auxílio na compra de produtos pela Internet tem um forte potencial e aceitação do público. Baseando-se nessas respostas, pode-se confirmar que a RA pode ser uma alternativa na visualização de produtos online.

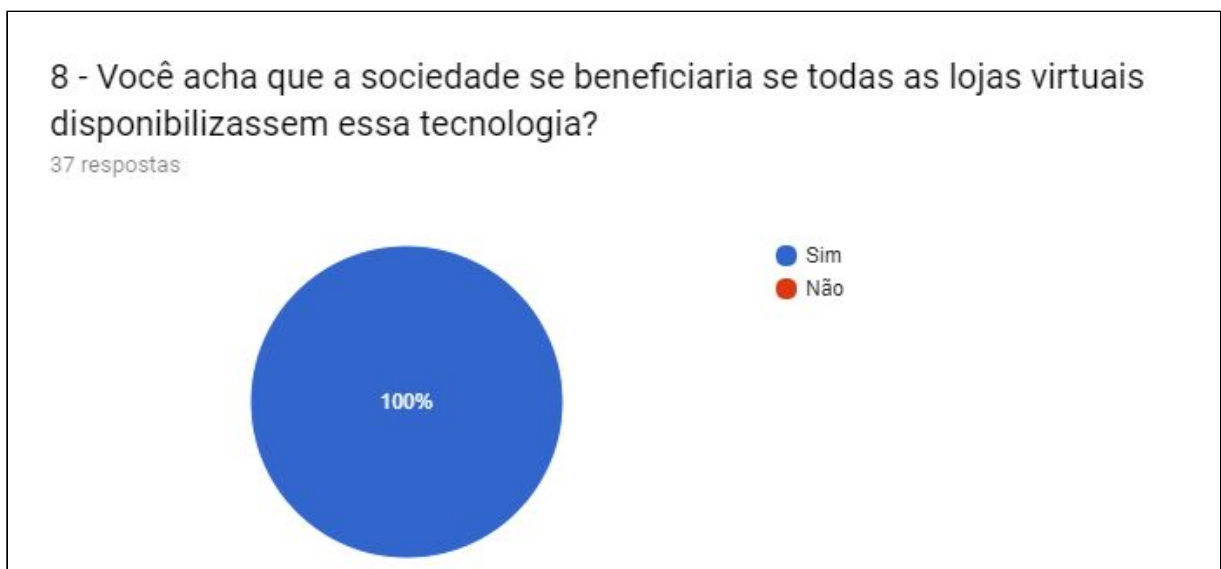


Figura 42 - Respostas da questão 8 do questionário.



## 5. CONCLUSÕES

O trabalho apresentou uma análise sobre a aplicação da Realidade Aumentada sem marcadores em dispositivos móveis, uma tecnologia não tão recente mas que ainda tem sido pouco utilizada pela sociedade. A análise dos questionários respondidos sugere uma necessidade do público em ter uma maneira mais interativa e de melhor visualização ao comprar um produto online, dado que todo o público que avaliou a aplicação achou útil poder visualizar um produto online, no seu ambiente de destino, antes mesmo de realizar a compra.

Os resultados da aplicação foram avaliados satisfatoriamente pelo público que a testou. Dessa maneira, a aplicação de Realidade Aumentada se mostrou como uma alternativa viável na visualização de produtos online, exibindo a utilização de uma tecnologia antes restrita, mas que está agora disponível facilmente para uso em dispositivos móveis, podendo inclusive ser utilizada em hardwares mais modestos. Entretanto, também pode-se concluir que se a tecnologia se tornar um padrão, ainda será necessário realizar melhorias para que esta se torne interoperável e mais realista, dado os problemas identificados nos resultados da questão 6 do questionário. Algumas melhorias propostas são: um formato padrão de arquivo para armazenar os modelos 3D junto com seu material, para que este possa ser utilizado independente de plataforma; e a necessidade do objeto receber luz e projetar sombra de acordo com a luz real do ambiente, pois isso aumentaria sua realidade, permitindo ser melhor inserido na cena.

Conclui-se então que o uso da Realidade Aumentada é capaz de possibilitar um avanço aos meios tradicionais (imagem e vídeo) de visualização de um produto online, permitindo que o consumidor, com a utilização dos dispositivos de celulares, possa ver o produto previamente em seu próprio ambiente, proporcionando mais confiança para compras online de produtos.

## REFERÊNCIAS

AMIN, Dhiraj; GOVILKAR, Sharvari. Comparative study of augmented reality sdk's. **International Journal on Computational Science & Applications**, v. 5, n. 1, p. 11-26, 2015.

ASUS, Zenfone. **Zenfone AR: Go Beyond Reality**. Disponível em: <<https://www.asus.com/Phone/ZenFone-AR-ZS571KL/>>. Acesso em: 16 maio 2017.

AUTODESK. **Maya**. Disponível em: <<https://www.autodesk.com.br/products/maya/overview>>. Acesso em: 02 jul. 2017.

AUTODESK, **3D Studio Max**. Disponível em: <<https://www.autodesk.com.br/products/3ds-max/overview>>. Acesso em: 02 jul. 2017.

ARAÚJO, Thúlio et al. **Life Cycle of a SLAM System: Implementation, Evaluation and Port to the Project Tango Device**. In: Virtual and Augmented Reality (SVR), 2016 XVIII Symposium on. IEEE, 2016. p. 10-19.

AZUMA, R., BAILLOT, Y., BEHRINGER, R., FEINER, S., JULIER, S., MACINTYRE, B. **Recent Advances in Augmented Reality**. 2001. IEEE Computers Graphics & Applications, vol. 21, no. 6, p. 34-47.

BASTOS, Nacha Costa; TEICHRIEB, Verônica; KELNER, Judith. **Interação com Realidade Virtual e Aumentada**. In: Symposium on Virtual Reality, VIII., 2006, Belém. Fundamentos e tecnologia de Realidade Virtual e aumentada. Porto Alegre: SBC, 2006. p. 129-148.

BLENDER, Blender Foundation. **Blender**. Disponível em: <<https://www.blender.org/>>. Acesso em: 01 jul. 2017.

BOOTSTRAP. **Bootstrap**. Disponível em: <<http://getbootstrap.com/>>. Acesso em: 02 jul. 2017.

BRAGA, Robson Ferreira. **Comparativo de Softwares de Autoria de Realidade Aumentada Almejando ser usado por Educadores**. 2016. 25 p. Monografia (Graduação em Licenciatura Plena em Computação)- Universidade Estadual da Paraíba, Campina Grande, 2016.

BROLL, Wolfgang et al. **An infrastructure for realizing custom-tailored augmented reality user interfaces**. IEEE transactions on visualization and computer graphics, v. 11, n. 6, p. 722-733, 2005.

DAQRI. **ARToolKit**. Disponível em: <<https://www.artoolkit.org/>>. Acesso em: 02 jul. 2017.

D'ANGELO, Thiago. **Tecnologias HMD Para Realidade Aumentada**. Disponível em: <<http://www.decom.ufop.br/imobilis/tecnologias-hmd-para-realidade-aumentada-part-e-1/>>. Acesso em: 12 abr. 2017.

DUFFY, Bob. **The Case for Augmented Virtuality**. Disponível em: <<https://software.intel.com/en-us/blogs/2016/04/26/the-case-for-augmented-virtuality>>. Acesso em: 11 abr. 2017.

FACHINETTO, Marcelo . **A Realidade Aumentada como suporte ao e-commerce**. 2013. 74 p. Trabalho de conclusão de curso (Bacharel em Tecnologias Digitais) - Centro de Computação e Tecnologia da Informação, Universidade de Caxias do Sul, Caxias do Sul, 2013.

FERREIRA, Maria Alice Grigas Varella; PELLEGRINO, Sérgio Roberto Matiello. **Modelagem 3D**. In: Symposium on Virtual Reality, VIII., 2006, Belém. Fundamentos e tecnologia de Realidade Virtual e aumentada. Porto Alegre: SBC, 2006. p. 246-264.

GITHUB. **GitHub Development Platform**. Disponível em: <<https://github.com/>>. Acesso em: 30 jun. 2017.

GONÇALVES, Rui Miguel Barros. **Juntando Modelação 3D Manual e Procedimental**. 2016.76 p. Dissertação (Mestrado Integrado em Engenharia Informática e Computação) - Faculdade de Engenharia, Universidade do Porto, Porto, 2016.

HIGA, Paulo. **Tudo o que você precisa saber sobre o Snapdragon 835**. Disponível em: <<https://tecnoblog.net/205642/snapdragon-835-tudo-sobre/>>. Acesso em: 15 maio 2017.

JACOBSON, Linda. **Realidade virtual em casa**. Tradução de PIRES, S.B.C Rio de Janeiro: Berkeley Press, 446 pp, 1994.

KIRNER, Claudio; KIRNER, Tereza Gonçalves. **Evolução e Tendências da Realidade Virtual e da Realidade Aumentada**. In: Simpósio de Realidade Virtual e Aumentada, XIII., 2011, Uberlândia, MG. Realidade Virtual e Aumentada: Aplicações e Tendências... Sociedade Brasileira de Computação - SBC, 2011. p. 10-25.

KIRNER, Claudio; PINHO, Márcio S.; **Uma Introdução à Realidade Virtual**. Mini-curso, X Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, Campos do Jordão - SP, 1997.

KIRNER, Cláudio; SISCOOTTO, Robson. **Realidade Virtual e Aumentada: Conceitos, Projeto e Aplicações**. Livro do pré-simpósio, IX Symposium on Virtual and Augmented Reality. Petrópolis, RJ: SBC – Sociedade Brasileira de Computação, 2007.

KLEIN, Georg; MURRAY, David. Parallel tracking and mapping on a camera phone. In: **Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on**. IEEE, 2009. p. 83-86.

KUDAN. **Kudan: Computer Vision**. Disponível em: <<https://www.kudan.eu/>>. Acesso em: 02 jul. 2017.

LIMA, João Paulo Silva do monte et al. **Study about natural feature tracking for augmented reality applications on mobile devices**. In: Virtual and Augmented Reality (SVR), 2015 XVII Symposium on. IEEE, 2015. p. 7-14.

LIMA, João Paulo Silva do Monte. **Realidade Aumentada Sem Marcadores Multiplataforma Utilizando Rastreamento Baseado em Modelo**. 2010. 85 f. Dissertação (Mestrado em Ciência da Computação) - Centro de Informática, Universidade Federal de Pernambuco, Recife, 2010.

LIMA, J., Apolinário, E., Farias, T., Silva, M., Teichrieb, V., & Kelner, J. (2007). **A Survey of Online Monocular Markerless Augmented Reality**. *International Journal of Modeling and Simulation for the Petroleum Industry*, 1, 1-8.

LIMA, J., Simões, F., Figueiredo, L., Teichrieb, V., & Kelner, J. (2010). **Model Based Markerless 3D Tracking applied to Augmented Reality**. *Journal on 3D Interactive Systems*, 1.

MANSSOUR, Isabel Harb; COHEN, Marcelo. **Introdução à Computação Gráfica**. RITA, v. 13, n. 2, p. 43-68, 2006.

MILETTO, Evandro Manara; BERTAGNOLLI, Silvia de castro . **Desenvolvimento de Software II: Introdução ao Desenvolvimento Web com HTML, CSS, JavaScript e PHP** - Eixo: Informação e Comunicação-Série Tekne. Bookman Editora, 2014.

MILGRAM, Paul et al. **Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum**. In: Telemanipulator and Telepresence Technologies. Boston, SPIE, V.2351, pp. 282-292, Outubro, 1994.

MICROSOFT. **Visual Studio Code**. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 01 jul. 2017.

MULTIDOTS. **Augmented Reality**. Disponível em: <<http://www.multidots.com/augmented-reality/>>. Acesso em: 30 abr. 2017.

NETTO, Antonio valerio; MACHADO, Liliane dos Santos; OLIVEIRA, Maria Cristina Ferreira de. **Realidade virtual-definições, dispositivos e aplicações**. Revista Eletrônica de Iniciação Científica-REIC. Ano II, v. 2, 2002.

NEWCOMBE, Richard A.; DAVISON, Andrew J. Live dense reconstruction with a single moving camera. In: **Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on**. IEEE, 2010. p. 1498-1505.

NISTÉR, David. (2003). **Preemptive RANSAC for live structure and motion estimation**. Proceedings Ninth IEEE International Conference on Computer Vision, (Iccv), 199-206 vol.1. Ieee. doi: 10.1109/ICCV.2003.1238341.

NOAH, Snaveley; SEITZ, Steven M.; SZELISKI, Richard. **Photo Tourism: Exploring image collections in 3D**. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2006)*, 2006.

OLIVEIRA, Lucas de Araújo. **Desenvolvimento do NetARToolkit: Um Sistema Distribuído de Realidade Aumentada**. 106 p. Dissertação – Mestrado em Ciência da Computação. Faculdade de Ciências Exatas e da Natureza. Universidade Metodista de Piracicaba. Piracicaba - SP, 2008.

OLIVEIRA, Filipe. **Vendas na Internet no Brasil devem dobrar até 2021, indica Google**. Disponível em: <<http://www1.folha.uol.com.br/mercado/2016/10/1823568-vendas-na-Internet-devem-dobrar-ate-2021-indica-google.shtml>>. Acesso em: 04 abr. 2017.

OLSSON, Carl. **The Structure from Motion Pipeline**. Disponível em: <<http://www.maths.lth.se/matematiklth/personal/calle/>>. Acesso em: 20 abr. 2017.

OpenCV. **Tutorial Aruco Detection**. Disponível em: <[http://docs.opencv.org/3.1.0/d5/dae/tutorial\\_aruco\\_detection.html](http://docs.opencv.org/3.1.0/d5/dae/tutorial_aruco_detection.html)>. Acesso em: 30 abr. 2017.

PARKIN, Simon. **After the success of Pokémon Go!, what is the future for augmented reality?**. 2016. Disponível em: <<https://www.theguardian.com/technology/2016/oct/23/augmented-reality-development-future-smartphone>>. Acesso em: 03 maio 2017.

PARENTE, André. **O Virtual e o Hipertextual: A rede como paradigma da contemporaneidade**. Rio de Janeiro, 1999. Disponível em: <<http://pt.scribd.com/doc/19455767/O-Virtual-e-O-Hipertextual-Andre-Parente>>. Acessado em 02/04/2017.

PEDDIE, Jon. **Augmented Reality: Where We Will All Live**. Springer, 2017.

PEREIRA, Ivo; NOGUEIRA, Nuno. **Realidade Virtual**. Disponível em: <<http://web.ist.utl.pt/ist170613/>>. Acesso em: 01 jul. 2017.

PIMENTEL, K.; TEIXEIRA, K. **Virtual Reality – through the new looking glass**. 2a edição. Nova York: McGraw-Hill, 1995. 438 pp.

POLLEFEYS, M., Nistér, D, Frahm, J. M., Akbarzadeh, A., Mordohai, P., Clipp, B., et al. (2007). **Detailed Real-Time Urban 3D Reconstruction from Video**. International Journal of Computer Vision, 78(2-3), 143-167. Springer. doi: 10.1007/s11263-007-0086-4.

POWER, Rachael. **VR-optimised Android Nougat starts to roll out**. Disponível em: <<https://www.virtualreality-news.net/news/2016/aug/23/vr-optimised-android-nougat-70-starts-roll-out/>>. Acesso em: 16 maio 2017.

ROBERTO, Rafael; LIMA, João Paulo; TEICHRIEB, Veronica. **Tracking for mobile devices: A systematic mapping study**. Computers & Graphics, v. 56, p. 20-30, 2016.

RAMOS, Durval. **Pokémon GO e o futuro da Realidade Aumentada**. Disponível em: <<http://newgameplus.com.br/pokemon-go-e-o-futuro-da-realidade-aumentada/>>. Acesso em: 15 maio 2017.

SAÇASHIMA, Rosemary Emika. **A Realidade Aumentada: desafios técnicos e algumas aplicações em jogos e nas artes visuais**. 2011. 90p. Dissertação (Mestrado em Artes), Universidade de São Paulo, São Paulo, 2011.

SIMÕES, Francisco Paulo Magalhães. **Realidade aumentada sem marcadores a partir de rastreamento baseado em textura uma abordagem baseada em pontos de interesse e filtro de partículas**. 2011. 91 f. Dissertação (Mestrado em Ciência da Computação) - Centro de Informática, Universidade Federal de Pernambuco, Recife, 2011.

SPURLOCK, Jake. **Bootstrap: Responsive Web development**. " O'Reilly Media, Inc.", 2013.

STEUER, Jonathan. **Defining virtual reality: Dimensions determining telepresence**. Journal of communication, v. 42, n. 4, p. 73-93, 1992.

II Encontro de Ferrovias. **Sistemas de Realidade Virtual para Simulação de Equipamentos de Movimentação e Treinamento de Operadores**, 2010, Vitória. Anais do II Encontro de Ferrovias. São Paulo: OTM Editora, 2010. v. 1. p. 1-2.

TORI, Romero; KIRNER, Claudio. **Fundamentos de Realidade Virtual e Aumentada**. In: Symposium on Virtual Reality, VIII., 2006, Belém. Fundamentos e tecnologia de Realidade Virtual e aumentada. Porto Alegre: SBC, 2006. p. 2-38.

UNITY. **Unity 3D**. Disponível em: <<https://unity3d.com>>. Acesso em: 02 jul. 2017.

UNITY. **Unity Manual - AssetBundles**. Disponível em: <<https://docs.unity3d.com/Manual/AssetBundlesIntro.html>>. Acesso em: 02 jul. 2017.

USP, Universidade de São Paulo. **Simulador de Trem**. Disponível em: <[http://www.usp.br/ldsv/?page\\_id=87](http://www.usp.br/ldsv/?page_id=87)>. Acesso em: 15 maio 2017.

VIALI, Lorí. **Elaborando um Questionário**. Disponível em: <[http://www.pucrs.br/famat/viali/mestrado/mqp/material/laminas/Educem\\_4.pdf](http://www.pucrs.br/famat/viali/mestrado/mqp/material/laminas/Educem_4.pdf)>. Acesso em: 01 nov. 2017.

WERNECK, Nicolau Leal; COSTA, Anna Helena Reali. SLAM monocular com reconstrução de planos para ambientes internos. In: **Workshop on MSc Dissertation and PhD Thesis in Artificial Intelligence**. 2010.

YOUNG, David. **Myron Krueger**. Disponível em: <<http://www.inventinginteractive.com/2010/03/22/myron-krueger/>>. Acesso em: 15 maio 2017.

ZORZAL, Ezequiel Roberto. **Estratégia para o Desenvolvimento de Aplicações Adaptativas de Visualização de Informações com Realidade Aumentada**. Uberlândia: UFU, 2009. 192 p. Tese (Doutorado) – Pós Graduação em Engenharia Elétrica, Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, Uberlândia, 2009.

## APÊNDICES

### APÊNDICE I - MÉTODO DE DOWNLOAD DO MODELO 3D

```
// rotina de download assíncrona
IEnumerator AsyncDownloadModel (string url)
{
    Debug.Log(url);
    // habilita a imagem animada de download
    downloadImage.enabled = true;
    // faz uma requisicao web para determinada url
    UnityWebRequest request = UnityWebRequest.GetAssetBundle(url, 0);
    request.Send();

    txtResult.text = "Downloading...";
    while (!request.isDone) {
        // exibe o resultado o progresso do download
        txtResult.text = Mathf.Round(request.downloadProgress * 100).ToString()+"%";
        yield return null;
    }
    // se acontecer algum erro
    if (request.isError) {
        Debug.Log(request.error);
        txtResult.text = "Não foi possível obter o modelo 3D";
        downloadImage.enabled = false;
    } else {
        // carrega o assetbundle
        AssetBundle bundle = DownloadHandlerAssetBundle.GetContent(request);

        // se carregado corretamente
        if (bundle) {
            txtResult.text = "Download concluído...";
            // obtem o objeto de nome model do asset bundle
            StaticData.objeto = bundle.LoadAsset<GameObject>("model");
            downloadImage.enabled = false;

            // se já estava pré visualizando um objeto
            if (previewObject)
                // destroi o objento
                Destroy (previewObject);
            // chama o método responsável pela pré-visualização
            PreviewModel ();
            //evita um erro ao tentar importar o mesmo asset bundle mais de uma vez
            bundle.Unload (false);
        } else {
            txtResult.text = "Não foi possível obter o modelo 3D";
        }
    }
}
```

## APÊNDICE II - PRINCIPAIS FUNÇÕES DO OBJETO RA

```
// metodo chamado pelo slider de altura, seta a altura do "piso virtual" do kudan,
// só funciona com o rastreamento parado
public void Altura(float altura){

    // se estiver rastreado para o rastreio
    if (_kudanTracker.ArbiTrackIsTracking ()) {
        _kudanTracker.ArbiTrackStop ();
    }
    // seta essa altura
    _kudanTracker.SetArbiTrackFloorHeight (altura * 100);

    // exhibe no txtAltura
    txtAltura.text = Mathf.Round (altura * 100).ToString();

    //Debug.Log (altura * 100);
}

// metodo chamado pelo icone "olho",
// inicia ou para o rastreamento dependendo do estado
public void Rastrear()
{
    if (!_kudanTracker.ArbiTrackIsTracking()) {
        // baseado no icone de posicionamento (Arrow)
        Vector3 floorPosition;
        Quaternion floorOrientation;

        // obtem o posicionamento e orientacao do "piso" da cena
        _kudanTracker.FloorPlaceGetPose(out floorPosition, out floorOrientation);
        // inicia o rastreamento sem marcador
        _kudanTracker.ArbiTrackStart(floorPosition, floorOrientation);
    } else {
        _kudanTracker.ArbiTrackStop();
    }
}
```



### APÊNDICE III - QUESTIONÁRIO QUALITATIVO

1 - Você já conhecia a realidade aumentada? \*

- ☐ Sim  
☐ Não

2 - No caso de já conhecer, você já teve contato com a realidade aumentada? (usou algum aplicativo ou dispositivo que fizesse uso de RA)

- ☐ Sim  
☐ Não

3 - Você já teve dúvidas ao comprar um produto pela internet e preferiu comprar este em uma loja física? \*

- ☐ Sim  
☐ Não

4 - O quão útil achou poder visualizar o produto no seu ambiente (na sua residência por exemplo) antes mesmo de finalizar a compra? \*

	1	2	3	4	
Pouco útil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito útil

5 - O aplicativo te ajudaria a tomar uma decisão de compra sobre um produto que está em dúvida (ex: cor, tamanho, local destinado)? \*

	1	2	3	4	
Não ajudaria	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Ajudaria muito

6 - No caso de achar que o aplicativo não ajudaria na tomada de decisão, explique o motivo principal.

- ☐ Falta de realismo do objeto virtual  
☐ Falta de precisão no tamanho do objeto  
☐ Complexidade de uso da aplicação  
☐ Complexidade para posicionar o objeto  
☐ Outro: \_\_\_\_\_

7 - Caso fosse dono de uma loja virtual, acha que os usuários se sentiriam mais confortáveis ao usar o aplicativo e que isso poderia incrementar as vendas? \*

- ☐ Sim  
☐ Não

8 - Você acha que a sociedade se beneficiaria se todas as lojas virtuais disponibilizassem essa tecnologia? \*

- ☐ Sim  
☐ Não