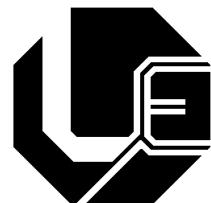


UNIVERSIDADE FEDERAL DE UBERLÂNDIA – UFU
FACULDADE DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



**Desenvolvimento de Sistema de
Geolocalização em Realidade
Aumentada para Multiplataforma
Móvel**

Gesmar de Paula Santos Júnior

Uberlândia
2015

Gesmar de Paula Santos Júnior

**Desenvolvimento de Sistema de
Geolocalização em Realidade
Aumentada para Multiplataforma
Móvel**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para a obtenção do título de Mestre em Ciências.

Área de concentração: Processamento da Informação, Computação Gráfica

Orientador: Alexandre Cardoso, Dr

Coorientador: Edgard A. Lamounier Júnior, PhD

Uberlândia
2015

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

S237d Santos Júnior, Gesmar de Paula, 1988-
2015 Desenvolvimento de sistema de geolocalização em realidade

aumentada para multiplataforma móvel / Gesmar de Paula Santos
Júnior. - 2015.

61 f. : il.

Orientador: Alexandre Cardoso.

Coorientador: Edgard Afonso Lamounier Júnior.

Dissertação (mestrado) - Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Elétrica.

Inclui bibliografia.

1. Engenharia elétrica - Teses.
 2. Realidade aumentada - Teses.
 3. Pórticos estruturais - Teses.
 4. HTML (Linguagem de programação de computador) - Teses.
- I. Cardoso, Alexandre, 1964-. II. Lamounier Júnior, Edgard Afonso, 1964-. III. Universidade Federal de Uberlândia, Programa de Pós-Graduação em Engenharia Elétrica. IV. Título.

CDU: 621.3

Gesmar de Paula Santos Júnior

**Desenvolvimento de Sistema de
Geolocalização em Realidade
Aumentada para Multiplataforma
Móvel**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para a obtenção do título de Mestre em Ciências.

Área de concentração: Processamento da Informação, Computação Gráfica

Uberlândia, 25 de Fevereiro de 2015

Banca Examinadora:

Alexandre Cardoso, Dr – FEELT/UFU

Edgard A. Lamounier Júnior, PhD –
FEELT/UFU

Claudio Kirner, Dr – IMC/UNIFEI

Keiji Yamanaka, PhD – FEELT/UFU

À minha família e aos meus amigos.

*“Não são as espécies mais fortes que sobrevivem nem as mais inteligentes,
e sim as mais suscetíveis a mudanças.”*
(Charles Darwin)

Resumo

Junior, G. S. **Desenvolvimento de Sistema de Geolocalização em Realidade Aumentada para Multiplataforma Móvel.** 61 p. Dissertação – Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, 2015 .

A Realidade Aumentada tornou-se tema popular no campo da pesquisa acadêmica, principalmente por permitir ao usuário uma visão de objetos virtuais sobrepostos ou compostos com o mundo real. Seu uso em dispositivos móveis constitui um campo de pesquisa em crescimento atualmente. Entretanto, uma das grandes dificuldades no desenvolvimento desta categoria de aplicativos é fato de existir diversas plataformas, com linguagens de programação, padrões e *SDKs* próprios. Este trabalho objetiva desenvolver uma aplicação multiplataforma com Realidade Aumentada baseada em geolocalização para navegação em ambientes abertos. Para tanto, foi utilizado o *PhoneGap*, um *framework* para a criação de aplicativos de plataformas móveis que usa *HTML5* e *JavaScript*.

Palavras-chave: Realidade Aumentada Móvel; *framework PhoneGap*; Geolocalização; *HTML5*.

Abstract

Junior, G. S. **Development Geolocation System in Augmented Reality for Mobile Multiplatform.** 61 p. Master Thesis – Faculty of Electrical Engineering, Federal University of Uberlândia, 2015 .

Augmented Reality became a popular topic within the field of academic research, especially by providing the user with an overlapping or compounding view of virtual objects and the real world. The use of Augmented Reality in mobile devices is currently growing as a field of research. However, one of the greatest difficulties about developing such application category is the fact that there are a great variety of platforms, programming languages, standards and proprietary SDKs. This work aims to develop a multi-platform outdoor navigation application using Augmented Reality based on geolocation. The application is then developed using PhoneGap, a framework platform for creating mobile applications that uses HTML5 and JavaScript.

Keywords: Mobile Augmented-reality, framework PhoneGap; Geolocation; HTML5.

List of illustrations

Figura 1 – Eventos, funções e sensores disponíveis nos dispositivos móveis.	20
Figura 2 – Componentes para aplicações da Realidade Aumentada Móvel	23
Figura 3 – Desafios da Realidade Aumentada Móvel	24
Figura 4 – <i>Equalizer augmented view</i>	33
Figura 5 – Funil de Atenção	34
Figura 6 – Ambiente de teste do Funil de Atenção	35
Figura 7 – Um <i>screenshot</i> de um dos vídeos utilizados no estudo	36
Figura 8 – <i>Screenshots</i> do <i>VisitAR application</i>	37
Figura 9 – Protótipo de aplicação baseado no <i>framework</i> do <i>HITLab</i>	38
Figura 10 – Esquema básico do Projeto <i>Basel</i>	39
Figura 11 – Caso de uso do sistema.	44
Figura 12 – Diagrama de sequência de utilizações do sistema.	48
Figura 13 – Tela inicial da aplicação apresentando o status do GPS e da bússola. .	52
Figura 14 – Tela do aplicativo.	52
Figura 15 – <i>App</i> em funcionamento.	53

Lista de tabelas

Tabela 1 – Comparações entre as ferramentas de autoria de Realidade Aumentada.	31
Tabela 2 – Comparações entre os trabalhos correlatos	40
Tabela 3 – Ator do sistema.	44
Tabela 4 – Descrição do caso de uso “Iniciar a aplicação”	45
Tabela 5 – Descrição do caso de uso “Visualizar Ambiente Aumentado”	45
Tabela 6 – Descrição do caso de uso “Visualizar Distância do destino”	46
Tabela 7 – Descrição do caso de uso “Visualizar Direção do destino”	46
Tabela 8 – Descrição do caso de uso “Seguir direção indicada”	47
Tabela 9 – Descrição do caso de uso “Fechar a Aplicação”	47

Lista de siglas

API *Application Programming Interface*

APP *Application*

BYOD *Bring Your Own Device*

GPS *Global Positioning System*

HTML5 *HyperText Markup Language*

IP *Internet Protocol*

ONU Organização das Nações Unidas

OS *Operational System*

RA Realidade Aumentada

RAM Realidade Aumentada Móvel

RV Realidade Virtual

SDK *Software Development Kit*

SO Sistemas Operacionais

W3C *World Wide Web Consortium*

Sumário

1	Introdução	12
1.1	Motivação	12
1.2	Objetivos	13
1.3	Metodologia	14
1.4	Organização da Dissertação	14
2	Fundamentos	15
2.1	Introdução	15
2.2	Sistemas Operacionais para Computação Móvel	15
2.2.1	<i>Windows Phone</i>	16
2.2.2	<i>Android</i>	17
2.2.3	<i>iOS</i>	17
2.3	Aplicações Híbridas	18
2.3.1	<i>PhoneGap</i>	19
2.4	Geolocalização	19
2.5	Realidade Virtual e Aumentada associada a dispositivos móveis	21
2.5.1	Justificativas para o Uso da Realidade Aumentada Móvel	22
2.5.2	Desafios da RA móvel	23
2.6	Bibliotecas de RA móvel	25
2.6.1	<i>Wikitude</i>	26
2.6.2	<i>Vuforia</i>	27
2.6.3	<i>Mixare</i>	28
2.6.4	<i>ARViewer</i>	28
2.6.5	<i>Layar</i>	29
2.6.6	Comparações	30
3	Trabalhos correlatos	32
3.1	Introdução	32

3.2	<i>Augmented Navigation in Outdoor Environments</i>	32
3.3	<i>Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms</i>	33
3.4	<i>A Mobile Indoor Navigation System Interface Adapted to Vision-Based Localization</i>	35
3.5	<i>VisitAR: a mobile application for tourism using AR</i>	36
3.6	<i>Building Mobile AR Applications Using the Outdoor AR Library</i>	38
3.7	<i>Mobile AR applications and services for use in the City of Basel</i>	39
3.8	Trabalhos Analisados	40
3.9	Conclusão	41
4	Especificação do Sistema Proposto	42
4.1	Introdução	42
4.2	Requisitos Funcionais	42
4.2.1	Aquisição de dados	42
4.2.2	Processamento de informações em tempo real	43
4.2.3	Criação do ambiente real aumentado	43
4.3	Requisitos não-funcionais	43
4.3.1	Requisitos de Implementação	43
4.3.2	Requisitos de Portabilidade	43
4.3.3	Requisitos de Eficiência	44
4.4	Caso de Uso	44
4.5	Diagrama de sequência	48
5	Detalhes da Implementação	49
5.1	Introdução	49
5.2	Obter a Posição do Usuário	49
5.3	Carregar a camada com informações de pontos de interesse	50
5.4	Cálculo da Distância entre Pontos	50
5.5	Cálculo da <i>azimute</i>	50
5.6	Bússola e <i>azimute</i>	51
5.7	Rosa dos ventos e distância do destino	51
6	Discussão dos resultados	54
6.1	Introdução	54
6.2	Analise Comparativa	54
Conclusão	55	
Referências	57	

CAPÍTULO 1

Introdução

1.1 Motivação

A Realidade Aumentada (RA) tornou-se tema popular no campo da pesquisa acadêmica, principalmente por permitir ao usuário uma visão de objetos virtuais sobrepostos ou compostos com o mundo real (CHANG; TAN, 2010), trazendo, assim, o enriquecimento de cenas de estudo. Na modelagem do mundo real, sistemas de RA devem prover funções que simulam as leis da física e, dependendo da forma com estas são implementadas, isso pode trazer sérias limitações à interação com o meio. Tal interação é facilitada por ferramentas de Realidade Aumentada, mediadas por dispositivos móveis como *tablets* e *smartphones*. Surge, assim, uma nova área de investigação científica e tecnológica: a Realidade Aumentada Móvel ou, simplesmente, RA Móvel.

A RA Móvel é uma das áreas de pesquisa que mais crescem no campo de Realidade Aumentada. Isso se deve principalmente ao surgimento de *smartphones*, que proporcionam um ambiente poderoso e onipresente para apoiar tal tecnologia (AZUMA et al., 2001).

Neste trabalho, as diferentes combinações de hardware e sistema operacional para dispositivos móveis são chamadas de plataformas móveis. Florins e Vanderdonckt (2004), ao discutir sobre interfaces gráficas em diferentes plataformas, chamam a atenção para o fato de que a transição de um aplicativo entre diferentes sistemas deve ser a mais suave possível.

Conforme discutido por Corral et al. (2011), uma das grandes dificuldades no desenvolvimento de aplicativos para dispositivos móveis deve-se ao fato de existir diversas plataformas, com linguagens de programação, padrões e *SDKs* próprios. Nestas condições, um mesmo aplicativo deve ter seu código fonte rescrito em linguagens aceitas por cada plataforma Java para aplicações *Android* e *Objective C* para *iOS*, por exemplo.

Uma maneira viável de se conseguir aplicações capazes de executar em diferentes plataformas é por meio do desenvolvimento web. Porém, conforme é discutido por Corral et al. (2011), esta não é uma tarefa simples, visto que, ao contrário do que ocorre em

aplicações *desktop*, em sistemas móveis existe uma variedade de recursos que devem ser tratados de forma diferente, como a orientação da tela, a interação por toque e o teclado virtual. Além disso, ainda segundo Corral et al. (2011), quando uma aplicação web é desenvolvida para ser acessada por *desktop*, toda parte da lógica reside do lado do servidor, diferentemente do que ocorre em aplicações móveis nativas, em que é preferível ter a no próprio dispositivo.

Ferramentas como o *Appcelerator*¹ ou *PhoneGap*² fornecem os meios para o desenvolvimento de aplicações com uma tecnologia baseada em *JavaScript*, *HTML* e *CSS*, permitindo que um único desenvolvimento possa ser implantado em diversas plataformas de destino. Porém, ainda não há à disposição da comunidade qualquer ferramenta multiplataforma voltada para o desenvolvimento de aplicações móveis com RA. A não existência de um *framework* capaz de gerar aplicativos de RA multiplataforma conduz a dois problemas frequentemente encontrados:

1. A dificuldade durante o desenvolvimento, pois são necessários conhecimentos em diferentes plataformas e tecnologias, gerando maior dificuldade e demandando maior tempo de desenvolvimento;
2. A maior demanda por tempo requerida ao desenvolvimento de várias versões de um mesmo aplicativo para diferentes plataformas reflete também na maior necessidade de tempo e mão de obra qualificada no suporte e manutenção dos aplicativos desenvolvidos, pois profissionais específicos de cada seguimento (plataforma) são necessários.

O desenvolvimento multiplataforma traz benefícios tanto para os usuários, que podem experimentar a mesma aplicação em diferentes ambientes, quanto para os desenvolvedores, por demandar menos esforço e menor tempo de trabalho durante o desenvolvimento e no processo de suporte e manutenção.

O aprendizado de RA Móvel tem sua curva melhorada por meio da utilização de padrões como o *HyperText Markup Language* (HTML5), definido pela *World Wide Web Consortium* (W3C), o que representa um ganho quando comparado aos *Software Development Kit* (SDK) tradicionais. O *framework* *PhoneGap*, utilizado neste trabalho, utiliza HTML5 e é um exemplo de ferramenta capaz de trazer tais benefícios.

1.2 Objetivos

Este trabalho objetiva superar os desafios no desenvolvimento de uma aplicação de realidade aumentada móvel para diferentes dispositivos e sistemas operacionais existentes

¹ <http://www.appcelerator.com/>

² <http://phonegap.com/>

por meio do desenvolvimento de uma aplicação de RA baseada em geolocalização para navegação *outdoor*, ou seja, em ambientes abertos. Tal desenvolvimento será uma aplicação nativa (mas que utiliza linguagens web) capaz de interagir com recursos disponíveis no dispositivo onde ela é executada por meio um *framework* conhecido como *PhoneGap*, devidamente detalhado ao longo deste trabalho.

Para cumprir tais requisitos, uma aplicação como esta deve apresentar as seguintes características:

1. Recursos de geolocalização;
2. Camadas de informações sobre pontos de interesse;
3. Realidade aumentada;
4. Multiplataforma.

1.3 Metodologia

A metodologia de desenvolvimento do presente trabalho foi orientada por três etapas principais:

1. A apresentação das principais e mais recentes referências de Realidade Aumentada Móvel encontradas na literatura e a avaliação dessa tecnologia;
2. Apresentação da aplicabilidade da RA móvel por meio de exemplos de aplicação, distinguindo, assim, elementos relevantes ao processo de desenvolvimento;
3. A análise e utilização dos diferentes recursos disponíveis nos dispositivos móveis (câmera, GPS, bússola, etc) para aprimorar a aplicação e apresentá-los em um estudo de caso.

Com esses conceitos e tecnologias, foi possível criar um sistema de navegação outdoor que direciona o usuário a determinados locais via indicação de dados geográficos utilizando *Global Positioning System* (GPS) e bússola. Com a inserção de objetos e informações alinhados com os do mundo real na tela do dispositivo com auxílio da câmera, foi criada uma aplicação de RA para navegação que pode ser utilizada em plataformas móveis existentes no mercado.

1.4 Organização da Dissertação

Essa dissertação está organizada em sete capítulos. O Capítulo 2 apresenta os principais fundamentos necessários ao entendimento deste trabalho. O Capítulo 3 apresenta

trabalhos correlatos com a presente pesquisa. O Capítulo 4 fornece a especificação do sistema proposto nesta dissertação, cujos detalhes de implementação são descritos no Capítulo 5. Os resultados são apresentados e discutidos no Capítulo 6, seguido pelo Capítulo de onde as Conclusões são apresentadas.

CAPÍTULO 2

Fundamentos

2.1 Introdução

Neste capítulo serão abordados tópicos que fundamentam a pesquisa sobre computação móvel, geolocalização e Realidades Virtual e Aumentada.

2.2 Sistemas Operacionais para Computação Móvel

Um Sistemas Operacionais (SO) (ou *Operational System* (OS)), é um dispositivo de software cujo trabalho é fornecer aos programas do usuário um modelo de computador melhor, mais simples e mais limpo e lidar com o gerenciamento de todos os recursos de hardware (TANENBAUM, 2003). Este conceito, aplicado aos computadores, é estendida aos dispositivos móveis atuais, que necessitam de um sistema operacional para gerenciar os recursos de hardware e software para dispositivos móveis e oferecer uma interface ao usuário final (SILBERSCHATZ; GALVIN; GAGNE, 2004).

Os *smartphones* agregam funcionalidades em seus hardwares e softwares, possuem capacidade de comunicação com redes de computadores ou até mesmo com outros dispositivos móveis e têm demonstrando grande sucesso na computação móvel (COSTA; FILHO, 2013). Por essa razão, importantes empresas provedoras de SO para computadores pessoais têm também apostado em sistemas para dispositivos móveis, principalmente *smartphones* e *tablets*. O contínuo surgimento de novos aparelhos impulsiona o desenvolvimento de novos SO. De fato, vários sistemas operacionais para esses têm sido desenvolvidos e buscam sua parcela de mercado, mas os que mais se destacam na atualidade são: *Android*, *iOS* e *Windows Phone*.

A maioria dos sistemas operacionais para *smartphones* é voltada para desenvolvimento por terceiros, permitindo a qualquer pessoa desenvolver aplicações através de um SDK ou *Framework* (conjunto de classes que colaboram para realizar o desenvolvimento de um subsistema ou da aplicação).

Embora facilitado por ferramentas SDK e *Framework*, o desenvolvimento de aplicações móveis é ainda fruto de investimentos de tempo e dinheiro. Por essa razão, espera-se que um aplicativo, depois de pronto, tenha ampla penetração no mercado consumidor, gerando lucros para a equipe de desenvolvimento. Disponibilizar a aplicação para sistemas operacionais consolidados e com boas previsões de permanência de mercado a médio e longo prazos traz maior segurança em termos de vida útil (MORIMOTO, 2009).

A falta de padronização que se verifica na grande maioria dos sistemas operacionais móveis é fator que acarreta problemas para empresas e usuários. A escolha do SO para o qual os aplicativos são desenvolvidos é frequentemente feita sem as devidas análises e sem o conhecimento prévio das arquiteturas e suas principais funcionalidades (COSTA; FILHO, 2013).

Assim, uma visão geral dos principais OS móveis faz-se necessária. Por essa razão, as próximas seções apresentam as principais características de três sistemas que mais se destacam no mercado atual: *Windows Phone*, *Android* e *iOs*.

2.2.1 *Windows Phone*

O *Windows Phone* foi criado com o intuito de ser uma nova plataforma para aplicativos da *Microsoft*. Um ponto de destaque é a possibilidade de qualquer desenvolvedor criar e compartilhar softwares instaláveis, aplicações para usuários finais, utilitários ou extensões para serviços. A empresa *Nokia* utiliza-o como SO padrão. Outras empresas, como a *Samsung*, *HTC* e *Motorola*, também comercializam aparelhos com *Windows Phone*.

De acordo com Bright (2010), para poder executar o *Windows Phone* um aparelho deve apresentar as seguintes especificações mínimas:

- ❑ Tela *multitouch* – interface de manipulação
- ❑ Botão *Back* – para sair dos aplicativos e para alternar entre os aplicativos em execução
- ❑ Botão *Start* – responsável por ligar o telefone e retornar à tela principal
- ❑ Botão *Search* – responsável pelo recurso de pesquisa interna ao programa;
- ❑ Câmera com de 5 (cinco) megapixels e *flash*;
- ❑ Acelerômetro – responsável por detectar a aceleração do telefone, o que representa uma mudança de velocidade da física, podendo também ser utilizado para detectar a orientação do telefone e/ou movimentos realizados com ele;
- ❑ *Compass* – responsável por detectar a orientação relativa ao Norte Magnético;
- ❑ *Location* – utilizado para detectar a localização geográfica do telefone, utiliza GPS;

2.2.2 *Android*

O sistema operacional *Android* é a plataforma de desenvolvimento para aplicativos móveis da empresa *Google*. Utiliza o *kernel* do *Linux* (versão 2.6), uma interface visual rica e traz um ambiente de desenvolvimento bastante poderoso, inovador e flexível (RUBIN, 2010). Atualmente, encontra-se na versão 4.4 *KitKat*.

Android é considerada a plataforma móvel mais popular atualmente e possibilita, em aparelhos móveis, a utilização dos aplicativos do *Google* disponíveis para computadores pessoais, *apps* e jogos. Os dispositivos *Android* são também considerados inteligentes, mantendo o foco no usuário, colocando-o no controle da experiência em mobilidade (INC, 2013). Fabricantes como *Acer*, *HP*, *HTC*, *Motorola* e *Sony Ericson* adotam o SO em seus aparelhos.

O *Android* é um sistema de código aberto, o que permite aos desenvolvedores o máximo aproveitamento de suas funcionalidades ao criar novos *Application* (APP). A loja virtual de aplicativos do *Google*, a *Google Play*, mantém disponíveis não somente os aplicativos oficiais criados pela empresa, mas também *apps* criados por desenvolvedores cadastrados. Os APP criados por terceiros podem ser gratuitos ou pagos, desde que se respeite a política de criação da empresa. A empresa *Google* disponibiliza também *Application Programming Interface* (API) para o desenvolvimento de novos recursos. Existem também ferramentas para a criação de aplicativos simples voltadas para não programadores. Como consequência, o SO torna-se ainda mais acessível, por oferecer ao usuário comum a possibilidade de criar seus próprios aplicativos.

Dentre os vários atrativos apresentados pela plataforma *Android*, destacam-se:

- ❑ Possui um navegador baseado no *framework* de Código aberto – *WebKit*;
- ❑ Utilização da máquina virtual *Dalvik*. Com ela, aplicações escritas em Java são compiladas para uso em dispositivos móveis, permitindo que programas possam ser executados em qualquer dispositivo *Android*, independentemente do processador utilizado;
- ❑ Suporte Adicional de Hardware, com o uso de câmeras de vídeo, tela sensível ao toque, GPS, acelerômetros, e aceleração de gráficos 3D;
- ❑ Ambiente de desenvolvimento (SDK) que inclui emulador, ferramentas de teste, memória e análise de desempenho.

2.2.3 *iOS*

O *iOS* é o sistema operacional móvel da *Apple*, derivado do *Mac OS*. Tendo sido primeiramente utilizado no aparelho *Iphone*, logo foi adotado também em outros aparelhos

da empresa (APPLE, 2014). Sua utilização é restrita aos aparelhos da *Apple*, sendo vedada a terceiros e sua interface é baseada no conceito de manipulação direta com multitoque.

A usabilidade, grande característica dos sistemas desenvolvidos pela *Apple*, é conseguida por meio da simplicidade do design. “O principal no nosso design é que temos de fazer as coisas intuitivamente óbvias” (ISAACSON, 2011). Com tal simplicidade, o usuário tende a descobrir naturalmente as funções do sistema, que são desenhadas de forma simples e clara.

A *Apple Store*, loja de aplicativos da *Apple*, conta também com grande quantidade de *Apps* gratuitos e/ou pagos. Quando comparado à plataforma *Android*, o desenvolvimento de aplicativos para *iOS* é mais limitado, devido principalmente aos direitos da empresa sobre seu sistema operacional. Este desenvolvimento está condicionado ao uso de softwares exclusivos e, oficialmente, o uso do sistema operacional para *desktop MAC OS*. Uma linguagem específica, com bases na linguagem C e chamada *Objective-C*, foi criada para a programação de aplicativos do *iOS*.

2.3 Aplicações Híbridas

Com a popularização dos dispositivos móveis, a demanda por aplicações para *smartphones*, *tablets*, etc. tem crescido consideravelmente. Esse contexto leva os desenvolvedores a enfrentar uma complexa questão: existe uma grande diversidade de aparelhos – sejam eles *tablets* ou *smartphones* – e eles executam sob diferentes sistemas operacionais e isso representa um grande desafio, já que é necessário o conhecimento aprofundado em diferentes plataformas para poder se alcançar maior fatia de mercado.

O custo para se treinar e manter equipes distintas com especializações em cada plataforma é alto. Isso se deve tanto às questões de quantidade de pessoal, treinamento e, principalmente de retrabalho: um mesmo aplicativo deve ser “traduzido” para diferentes plataformas. Os custos também são transferidos ao que se refere à manutenção. Uma aplicação pode apresentar comportamentos falhos em uma plataforma, mas não em outra, por exemplo. Assim, sob a homologação de cada aplicação incide um custo relativo ao tempo e expertise por parte da(s) equipe(s) de desenvolvimento.

Uma abordagem para se contornar esses problemas é o conceito de soluções híbridas. A ideia é que se elabore um código único e no final, gerar uma aplicação para cada plataforma, sem alterar seu código ou com alterações mínimas. Algumas plataformas possibilitam inclusive gerar aplicações *desktop* e sistemas web. Exemplos de *frameworks* que trabalham neste sentido são o *Titanium*, o *Cortona* e a *Adobe*, que utilizam o *Flex SDK* e o *Adobe Air*.

O uso de aplicações híbridas possibilita o desenvolvimento de aplicativos para vários ambientes sem a necessidade de se codificar para cada um deles. Há ainda a possibilidade de se utilizar código nativo em conjunto, caso haja algo que exija tal solução, aumentando

ainda mais a flexibilidade.

De acordo com o Gartner (2013) empresa de pesquisa e aconselhamento sobre tecnologia, em 2016, mais de 50% dos aplicativos móveis serão produzidos com sistemas híbridos, possibilitando, assim, um equilíbrio entre *Web Apps* (Aplicativos Web), baseados em HTML5 e aplicações nativas. Embora o desenvolvimento de aplicativos nativos ofereça melhor experiência aos usuários e desempenho para aplicativos móveis, o desenvolvimento em várias plataformas muitas vezes gera um conjunto fragmentado de ferramentas de desenvolvimento ou ainda várias versões de uma aplicação para atender a mesma necessidade, dependendo do tipo de dispositivo ou sistema operacional.

O HTML5 apresenta vários recursos, como a capacidade de executar a aplicação mesmo sem conexão de dados e avançadas ferramentas de animação. Porém, seu funcionamento é condicionado à existência de um navegador ou componente que simule um navegador. Dessa forma, o desenvolvimento de um aplicativo em arquitetura híbrida torna-se mais interessante, uma vez que este será capaz de aproveitar tanto as vantagens do HTML5 quanto as capacidades do dispositivo onde este aplicativo é executado – como eventos e sensores.

A tendência *Bring Your Own Device* (BYOD), ou, “traga seu próprio dispositivo” e o aumento do uso de aplicativos móveis, fazem com que os desenvolvedores busquem formas de gerenciar um portfólio de arquiteturas de aplicativos móveis. Por poder utilizar linguagens multiplataforma em diferentes projetos, as arquiteturas híbridas apresentam menor custo.

2.3.1 *PhoneGap*

Para que uma aplicação móvel execute corretamente suas funções, pode ser necessário que esta acesse recursos do dispositivo em que está instalada – como eventos, armazenamento, mídia e sensores. Um forma de se obter tal acesso é a utilização de interfaces específicas para cada sistema operacional, construídas com padrões próprios.

Porém, por meio de *frameworks* como o *PhoneGap* uma aplicação pode ser portável para diferentes plataformas e preservar essa capacidade de interação com os dispositivos com o mínimo de alterações. Em muitos casos, é possível utilizar um único código para todas as plataformas. Na sua versão atual, o *PhoneGap* está disponível para *iOS*, *Android*, *Blackberry*, *Windows Phone*, *Amazon-fireos*, *Firefox OS*, *Ubuntu*, *Tizen* e *Windows 8*. A Figura 1 apresenta os principais recursos que são acessíveis via *PhoneGap*.

2.4 Geolocalização

Geolocalização é a arte de descobrir onde um usuário está localizado e, opcionalmente, compartilhar essa informação com outras pessoas e/ou aplicativos. Diferentes métodos

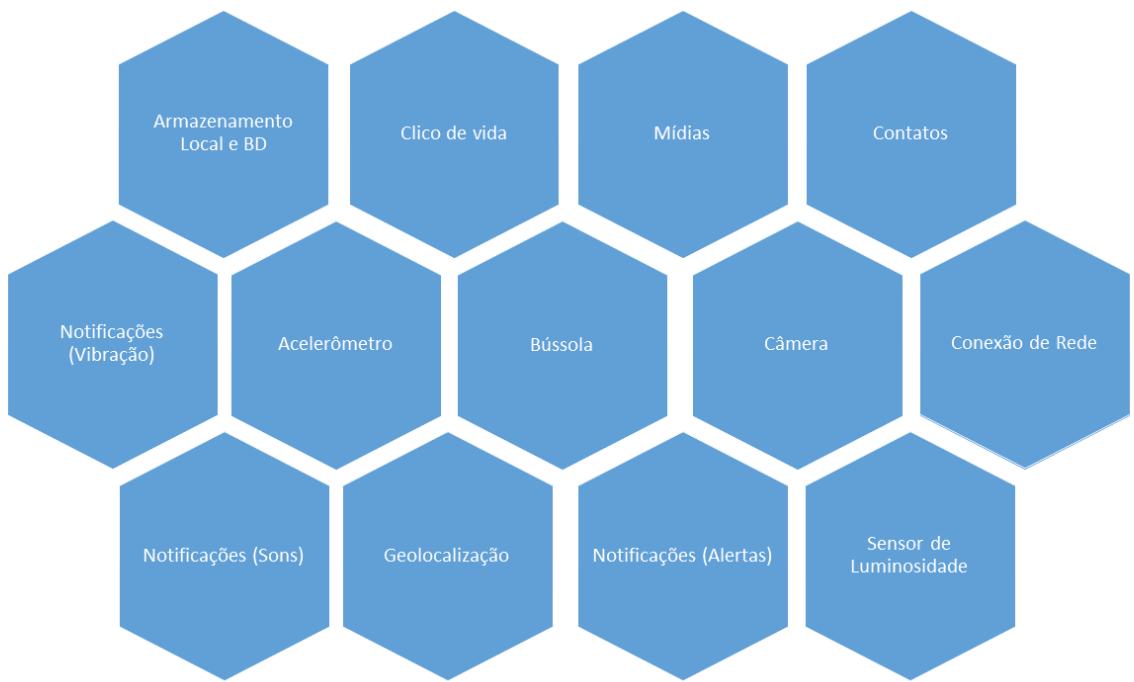


Figura 1 – Eventos, funções e sensores disponíveis nos dispositivos móveis.

podem ser utilizados para se descobrir a localização de um usuário: seu endereço *Internet Protocol* (IP), sua conexão de rede sem fio, a torre de celular que seu telefone está conectado, ou hardware GPS – dispositivo dedicado que calcula dados geográficos como latitude e longitude através de dados enviados por satélites.

Os primeiros métodos de descoberta da localização eram baseados no endereço IP, o que fornecia informações pouco confiáveis. Graças ao HTML5 e sua API de Geolocalização (*Geolocation*, em inglês), hoje é possível saber a posição do usuário de forma mais precisa, escrevendo aplicações que utilizam esse recurso.

A API de geolocalização permite ao usuário compartilhar sua localização em sites e aplicações confiáveis. As coordenadas são disponibilizadas na página via *JavaScript* que, por sua vez, pode ser enviado a um servidor e realizar ações como encontrar locais de interesse ao redor do usuário ou mostrar sua posição em um mapa.

Como pôde ser visto na Figura 1, a API de geolocalização é suportada pela maioria dos navegadores, tanto para *desktop* e quanto para dispositivos móveis. Além disso, vários navegadores e dispositivos antigos podem oferecer suporte à geolocalização via bibliotecas de terceiros.

A geolocalização mobile se destaca como a melhor plataforma com geolocalização para computação onipresente

2.5 Realidade Virtual e Aumentada associada a dispositivos móveis

A Realidade Virtual é uma interface computacional avançada que permite ao usuário interagir em tempo real com um ambiente tridimensional gerado por computador, usando seus sentidos, através de dispositivos especiais (KIRNER; KIRNER, 2011).

A Realidade Virtual (RV) trabalha com informações multisensoriais produzidas e manipuladas em tempo real. A interação em tempo real exige uma alta capacidade de processamento gráfico, sonoro e háptico. O uso de ambientes de RV exigem adaptação e treinamento para situar o usuário no mundo virtual.

Diferentemente da realidade virtual, que utiliza como meio de interação o ambiente virtual, a realidade aumentada mantém o usuário no seu ambiente físico e transporta o ambiente virtual para o mundo real, por meio de algum dispositivo tecnológico. Assim, a interação do usuário com os elementos virtuais ocorre de maneira natural e intuitiva, sem necessidade de adaptação ou treinamento.

De outra maneira, realidade aumentada pode ser definida como o enriquecimento do mundo real com informações virtuais geradas por computador em tempo real e devidamente posicionadas no espaço 3D, percebidas através de dispositivos tecnológicos (KIRNER; KIRNER, 2011).

Realidade Virtual e Aumentada são tecnologias dependentes de processamento em tempo real e, por isso, são influenciadas pela evolução da computação, tanto do ponto de vista do hardware quanto do software.

Para que os objetos virtuais façam parte do ambiente real e sejam manuseados deve-se utilizar um software com capacidade de visão do ambiente real e de posicionamento dos objetos virtuais, além de acionar dispositivos tecnológicos – hardware – apropriados para Realidade Aumentada.

O hardware de Realidade Aumentada pode usar dispositivos de Realidade Virtual, mas tende a não obstruir as mãos, que devem atuar naturalmente no ambiente misturado (TORI; KIRNER; SISCOUTTO, 2006).

Com a sobreposição de objetos virtuais, ao usuário é concedido acesso a informações que não são tipicamente disponíveis usando seus próprios sentidos. Naturalmente, esse ganho é ainda mais relevante quando utilizadas tecnologias de computação móvel, pois ao usuário é permitida a movimentação pelo ambiente, explorando informações novas e sua realidade simultaneamente (Sá; CHURCHILL, 2012).

Com o desenvolvimento de dispositivos que a cada dia apresentam melhorias de hardware, a evolução da interface com o usuário também se faz necessária. A Realidade Aumentada é uma área que, associada à comunicação móvel, permite ao usuário experiências interativas e dinâmicas na apresentação de um conteúdo (Sá; CHURCHILL, 2012).

O uso de RA móvel consiste basicamente de dois tipos diferentes de experiências:

geolocalização e realidade aumentada baseada em visão computacional. A RA baseada em geolocalização utiliza GPS, bússola e outros sensores no telefone celular do usuário para fornecer o software de navegação entre pontos de interesse. A RA baseada em visão computacional utiliza os sensores para exibir conteúdos digitais no contexto de objetos e marcadores do mundo real, como revistas, cartões postais ou embalagem de produtos, seguindo as características visuais desses objetos.

Com a crescente demanda por aplicações de RA para plataformas móveis, surge um novo campo de criação de ferramentas para localização especialmente construídas levando-se em conta as características e limitações inerentes a tais dispositivos, como o uso de sensores de orientação, GPS, Bússola e interfaces adaptadas à entrada por toque.

Os dispositivos móveis permitem o desenvolvimento de aplicações móveis de forma abrangente e dinâmica. Dentre as vantagens da mobilidade encontram-se o acesso a informação de forma rápida, flexível, em tempo real e de qualquer lugar. A área de pesquisa de aplicações móveis ainda está em sua “infância” (WOODILL, 2010).

2.5.1 Justificativas para o Uso da Realidade Aumentada Móvel

O mercado de celulares está crescendo cada vez mais e estudos apontam que mais de 3 bilhões de pessoas possuem um aparelho celular, o que corresponde aproximadamente à metade da população mundial (LECHETA, 2010). Os usuários estão procurando celulares e *smartphones* que possuem diversos recursos, tais como câmeras, *bluetooth*, jogos, interfaces de alta qualidade visual, GPS, acesso a internet e e-mails, entre outras aplicações. Sendo assim, a mobilidade está cada vez mais conhecida e utilizada, gerando demanda de novas aplicações.

Os dispositivos móveis podem ser ferramentas auxiliares à criação de experiências de aprendizagem e, associados a RA, tornarem-se estratégias de desenvolvimento em outras áreas, tais como jogos e visualização da informação.

Os principais benefícios do uso de dispositivos móveis para a RA incluem: a portabilidade, a mobilidade (estar disponível a qualquer hora e em qualquer lugar), acesso, de forma flexível, a diversos recursos e em tempo real, economia de tempo, rapidez da comunicação, capacitação e envolvimento dos alunos e o fato da aprendizagem móvel (*mobile learning*) ser abrangente e dinâmica (WOODILL, 2010).

De acordo com relatório da Organização das Nações Unidas (ONU), seis em cada dez pessoas no mundo têm assinatura de telefonia móvel, o que significa que os celulares representam a comunicação tecnológica mais escolhida nos últimos anos, especialmente nos países mais pobres (WOODILL, 2010).

Alguns fatores de mercado contribuem para efetivação de aplicativos e softwares com RA, uma vez que a diminuição do custo dos equipamentos que possui tecnologias adequadas para o desenvolvimento e utilização desses aplicativos se torna cada vez mais acessíveis nas diversas classes sociais. A amplitude em que a RA vem atuando se torna imprescindível.

dível e coerente para várias soluções, contribuindo assim para o desenvolvimento social e tecnológico.

No desenvolvimento de aplicações utilizando a Realidade Aumentada Móvel (RAM) para acessar ou armazenar informações devem ser considerados os componentes propostos na Figura 2. Dispositivos, tais como *smartphones* e *tablets*, são executados em uma infraestrutura de rede e são executadas em plataformas, tais como *Android* ou *iOS*. Nas plataformas são instaladas as ferramentas, as tecnologias e as bibliotecas que permitem que as aplicações accessem as fontes de informações (CARDOSO et al., 2012).

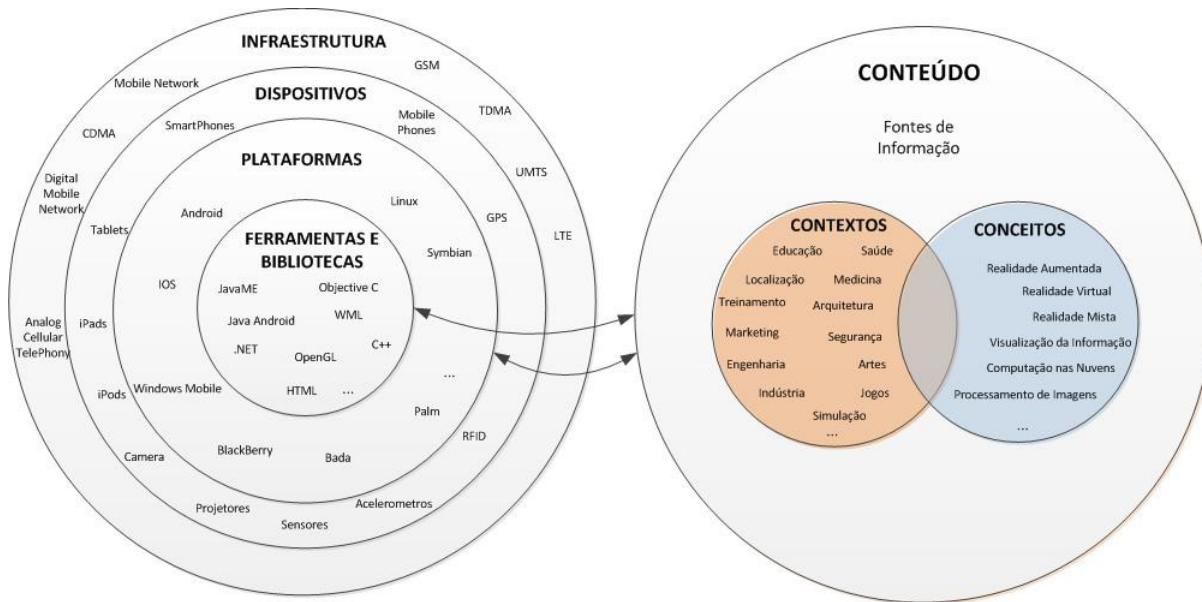


Figura 2 – Componentes para aplicações da Realidade Aumentada Móvel (CARDOSO et al., 2012).

O conteúdo inclui informações de contexto que são aplicações em diversas áreas, tais como Educação, Medicina, Jogos, Engenharia, entre outros, e os conceitos são as técnicas e formas que são usadas para acessá-los, tais como a Realidade Aumentada e a Visualização de Informação.

2.5.2 Desafios da RA móvel

Alguns desafios que devem ser considerados para a RAM são: o tamanho da tela, o consumo de energia, precisão de localização, câmera de qualidade, tempo de processamento, memória, registro, segurança, tempo real, cobertura, capacidade de processar grande quantidade de dados, dinamicidade, interação e conteúdo.

Estes desafios são mostrados na Figura 3, que classifica cinco categorias: hardware, aplicacão, rede, meio ambiente e utilizador.

Primeiramente, no desenvolvimento de uma aplicação móvel é necessário identificar quais são os recursos de hardware disponíveis em todos os dispositivos que se deseja atingir

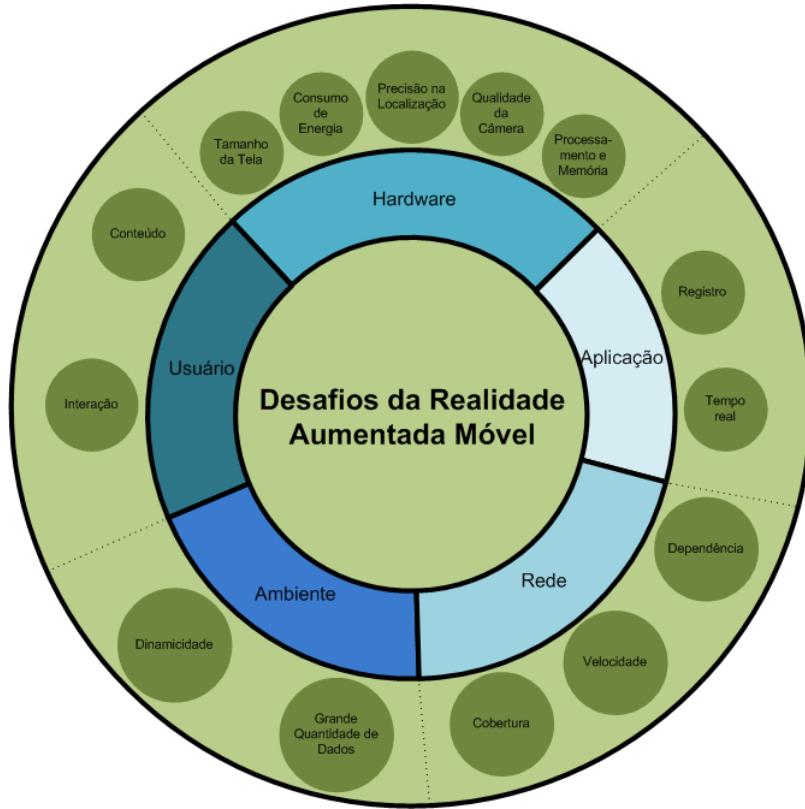


Figura 3 – Desafios da Realidade Aumentada Móvel (JUNIOR et al., 2013).

com a aplicação, como câmera e teclado, que estão presentes na maioria dos dispositivos existentes atualmente.

No entanto, para os outros tipos de sensores, não se pode garantir que todos devam funcionar ou existir em qualquer dispositivo móvel. Assim, é importante identificar o modelo de resposta dos sensores desenvolvidos e como eles serão utilizados no software, viabilizando o uso da aplicação mesmo com recursos limitados. Caso uma aplicação utilize um sensor que se encontra na maioria dos dispositivos, como bússola, giroscópio e GPS, existem algumas particularidades. Por exemplo, a precisão do GPS não é alta em ambientes *indoor* e alguns possuem câmera de baixa qualidade. Deve-se, então, tratar e apresentar a aplicação também nesses dispositivos de baixo desempenho, mesmo que seja de forma degradada ou apresentando uma precisão inferior, permitindo que o usuário execute a aplicação sem travamento ou que a aplicação fique aguardando a resposta a esses sensores.

A tela também pode dificultar o uso da aplicação, dado que aplicações desenvolvidas especificamente para telas de determinados tamanhos podem aparecer cortados e/ou distorcidos quando executados em dispositivos com telas de tamanhos diferentes. Portanto, para evitar complicações, deve-se trabalhar com *layouts* dinâmicos, diminuindo ícones, ocultando menus, trabalhando com diferentes configurações de tela, mantendo a aplicação funcionando em telas de menor tamanho e resolução.

Para dispositivos móveis, o hardware nem sempre apresenta alta duração de bateria, grande capacidade de processamento e memória, o que pode causar baixo desempenho, limitando as funcionalidades e, em um segundo momento, a utilização adequado de um dispositivo móvel.

Outro desafio se refere ao registro da aplicação, ou seja, se haverá controle de usuário, trazendo maior segurança com a devida autenticação da aplicação, quando disponível uma conexão de rede local ou internet. Essas necessidades devem ser avaliadas, de acordo com a disponibilidade da infraestrutura de servidores, arquitetura de hardware e software.

Um grande desafio é a questão de rede relacionada à velocidade de tráfego, área de cobertura das empresas de telefonia e o quanto uma dada aplicação é dependente dessa rede. Para RA *indoor*, esse problema pode ser minimizado, utilizando-se uma rede *wireless*. Porém, para ambientes do tipo *outdoor*, tem-se uma dependência das redes de telefonia.

O quinto desafio apresentado na Figura 3 refere-se ao usuário em si. O perfil do usuário pode exigir a disponibilização de elementos de auxílio (*Help*), além da necessidade de aprendizado das diferentes formas de interação com a solução de RA no dispositivo e a busca de novas informações. Outro quesito relevante refere-se à poluição visual que pode acontecer, por sobreposição de grande conjunto de informações visuais, por exemplo, sobre um elemento (ou posição) da cena real.

Todas essas questões acarretam em tempo e custo no projeto de desenvolvimento e devem ser analisadas e inseridas no projeto em fase de planejamento do software.

2.6 Bibliotecas de RA móvel

Existem diversas ferramentas que propõe o uso de Realidade Aumentada para dispositivos móveis. Nesta seção, são apresentadas as avaliações das principais bibliotecas que compõem a fundamentação teórica da pesquisa realizada no presente trabalho:

- ❑ *Wikitude*
- ❑ *Vuforia*
- ❑ *Mixare*
- ❑ *ARViewer*
- ❑ *Layar*

Ao desenvolver uma aplicação utilizando uma biblioteca, questões como licenciamento, suporte, acesso ao código fonte, compatibilidade e recursos podem viabilizar (ou mesmo, inviabilizar) o desenvolvimento. Por essa razão, a análise das bibliotecas foi realizada com base nas seguintes questões:

1. Software de código aberto ou disponíveis para uso gratuito?
2. Pode ser comercializado?
3. Pode ser incorporado em um aplicativo nativo para as plataformas móveis?
4. Apresenta as informações de geolocalização de um determinado ponto?
5. Existe uma comunidade de desenvolvimento?
6. Qual a quantidade e qualidade da documentação?
7. Quais são as opções de suporte disponíveis?
8. Suporte a conteúdo 3D?
9. Disponibilidade de funcionalidades equivalentes em diferentes plataformas, com suporte para migração ou desenvolvimento multiplataforma?

2.6.1 *Wikititude*

O *Wikititude* (SOUZA; BOTEGA, 2013) é um aplicativo de RA para navegação baseado em geolocalização. Com ele, pode-se encontrar serviços e produtos utilizando apenas com a câmera do dispositivo móvel. O software aplica sobreposições de Realidade Aumentada para exibição do conteúdo.

O *Wikititude* possui diversos serviços e sites integrados, a possibilidade de filtros de camadas ativas e inativas, além de serviços de busca. Essa ferramenta possui uma API disponível para desenvolvimento, com integração de linguagens multiplataforma (HTML5, CSS3 e *Javascript*) porém, até o momento, suporta apenas três plataformas: *iOS*, *Android* e *BlackBerry*.

O software não é *open source*, mas está disponível gratuitamente para projetos sem fins lucrativos. Porém, na versão gratuita a animação de inicialização do *Wikititude* é exibido um logotipo na visão da câmera, o que prejudica o ambiente de RA. Para evitar esse problema em versões comerciais, é necessário adquirir uma licença de uso, cujos valores variam de acordo com a quantidade de distribuições das aplicações desenvolvidas.

Dentre as vantagens apresentadas por essa API, destacam-se (MADDEN, 2011):

- ❑ Trabalha em conjunto com diversos recursos dos dispositivos móveis, como GPS, câmera e acelerômetros;
- ❑ Possui rastreamento baseado em marcadores e imagens capturadas pela câmera do dispositivo;
- ❑ Pode ser usado como uma ferramenta de pesquisa no mundo real, mostrando dados de lugares úteis, vídeos do *Youtube* ligados a locais reais, etc.

Um fator negativo refere-se à manipulação de objetos 3D. Embora esse recurso já esteja disponível para o *WebGL*, o que torna seu uso possível com HTML5, a manipulação 3D não está disponível no *Wikitude*.

Outra observação pertinente refere-se à comunidade. Embora o desenvolvimento do software siga ativamente, não há uma comunidade de apoio.

Além disso, relatos de usuários apontam o alto consumo de bateria verificado com o uso *Wikitude*, o que pode significar maior custo computacional.

Por fim, as camadas disponíveis nem sempre são úteis a todos usuários, e podem apresentar características muito distintas, criando uma certa falta de padrão.

2.6.2 *Vuforia*

Vuforia (NARDI; PERES,) é um *framework* de desenvolvimento realidade aumentada 3D disponível para as plataformas *Android* e *iOS*. Anteriormente conhecido como *Qualcomm QCAR*, o *Vuforia* é utilizado no reconhecimento no rastreamento (em tempo real) de imagens planares, alvos de imagem coloridas e objetos 3D simples – caixas, por exemplo.

Este *framework* destina-se ao desenvolvimento de RA baseada em visão da realidade especreto aumentada, que propõe reconhecimento de objetos do mundo real 2D e 3D e *multi-target* (mais conhecidos como *trackables*), configurações *multi-target 3D* e uma forma de marcador endereçável (conhecido como um marcador de quadro) com a sobreposição de modelos 3D no ambiente.

O software é proprietário, mas está disponível gratuitamente para uso em projetos comerciais e não-comerciais. É um *framework* que permite usar o reconhecimento de imagem no projeto e pode ainda ser usado como uma extensão para a plataforma *Wikitude* – caso se queria combinar Geolocalização e visão baseada em RA. Apesar das licenças gratuitas, toda aplicação desenvolvida coleta dados anonimamente e envia para a empresa que mantém a biblioteca, o que pode gerar incômodo e resistência por parte de alguns usuários.

A ferramenta está disponível apenas para as plataformas móveis *iOS* e *Android*, não se caracterizando, assim, como um sistema multiplataforma. Trabalha com conteúdo 3D de qualidade elevada, permitindo incluir efeitos de partículas, refração, normal *maps*, rastreamento através de imagens. É capaz de reconhecer objetos da câmera, mesmo que parcialmente escondidos (oculto), melhorando, dessa forma, a qualidade das aplicações. Permite também o uso de marcadores monocromáticos tradicionais.

Dentre as vantagens do *Vuforia*, destacam-se:

- ❑ A possibilidade de utilização de bases de dados remotas para diminuir a sobrecarga nos dispositivos móveis. Podem ser utilizadas tanto dados locais advindos de redes locais quanto de dados localizados em nuvens computacionais.

- ❑ A interatividade nas aplicações, como botões virtuais sobre objetos reais utilizados como marcadores.
- ❑ Possui um bom suporte ao desenvolvedor sendo mantido por uma empresa considerada sólida e bem estabelecida. Deve-se, no entanto, lembrar que a plataforma é recente, com menor popularidade e comunidade de desenvolvedores.
- ❑ Possui licenças gratuitas, mesmo para aplicações comerciais.

2.6.3 *Mixare*

Mixare Augmented Reality Engine (MIXARE, 2014) é uma biblioteca gratuita para RA. É um projeto *open source* para *Android* e *iOS* que fornece um aplicativo AR baseado em geolocalização. Inicialmente, já mostra os sites *Wikipedia*, Twitter e entradas em torno do usuário na visão por câmera (RA). Pode-se também ver através do mapa ou em uma lista. Esta ferramenta pode ser usada por outros aplicativos ou até mesmo por um site para exibir qualquer tipo de conteúdo. Além disso, é livremente expansível e pode até mesmo ser modificada em um aplicativo individual. A fonte de dados é transferida para a aplicação e acessada por um link em um site HTML, onde essa fonte é transferida para a aplicação.

O aplicativo é semelhante ao *Layar* (que será visto mais adiante) e ao *Wikitude*, mas difere destes, que fornecem sua infraestrutura aos desenvolvedores. O *Mixare* fornece apenas a *engine* de RA, que possibilita o desenvolvimento de uma aplicação nativa, sem necessidade de abrir o outro aplicativo para funcionar. O *Mixare* não possui permissão para desenvolvimento de aplicações comerciais, pois está restrito à licença *open-source*.

Este software trabalha com dados do GPS e acelerômetro para rastrear a posição e orientação do dispositivo e não possui opções de rastreamento via imagens ou marcadores.

Alguns aspectos negativos do *Mixare* referem-se à documentação (há pouca disponível), o fato de ser compatível somente com *Android* e *iOS* e oferecer apenas conteúdo 2D, trabalhando apenas com marcas estáticas e textos descritivos ligados a páginas com descrições dos objetos.

2.6.4 *ARViewer*

ARViewer (CONSULARO; COELHO; JR, 2007) é uma fonte *open-source* biblioteca de RA baseada em geolocalização para *Android* desenvolvida pela empresa *Arpa Solutions*. Uma das suas características mais importantes é a capacidade de usar a interface de RA para criar um conteúdo que é marcado com uma localização (latitude, longitude e altitude).

Para usar esta biblioteca, o usuário final precisa instalar o aplicativo *ARViewer* no dispositivo, bem como a aplicação desenvolvida. Esse procedimento pode ser complexo para a maioria dos usuários.

O *ARViewer* exibe conteúdo somente 2D e está disponível apenas na plataforma *Android*. Não trabalha com objetos 3D e conteúdos multimídia, apenas marcas estáticas e textos descritivos com links para páginas e conteúdos com descrições dos objetos. É semelhante ao *Layar* e *Wikitude*, porém pode ser usado como uma aplicação separada. Diferentemente do que ocorre com os demais, que fornecem a infraestrutura própria para uso dos desenvolvedores terceiros, o *ARViewer* fornece apenas um *browser*, o que traz vantagens no desenvolvimento de uma aplicação própria. Existe, porém, necessidade do aplicativo terceiro estar instalado no dispositivo para funcionar.

Este *framework* trabalha com dados do GPS e acelerômetro para rastrear a posição e orientação do dispositivo. Pode-se rastrear objetos pequenos usando *QRCodes* e exibir informações sobre eles. Ele pode também ser integrado a projetos *Android*, funcionando como um *framework* de auxílio em outras aplicações. É possível ainda inserir novos dados, e enviá-los para um servidor compartilhando com os demais usuários as novas informações.

A plataforma é recente, bastante simples em relação às outras e, até o momento da escrita dessa dissertação, não tem suporte ao *Windows Phone*, *iOS*, *BlackBerry* e demais plataformas. Devido à licença ser *open-source*, não está disponível para desenvolvimento de aplicações comerciais.

2.6.5 *Layar*

O *Layar* é uma ferramenta especializada em realidade aumentada móvel, sendo atualmente considerado o software mais popular entre os usuários de *smartphones* que integram com conteúdo de RA. Trata-se de um aplicativo proprietário, para as plataformas *iOS* e *Android* (MADDEN, 2011).

A arquitetura do *Layar* suporta a criação de várias camadas, com informações de geolocalização (mostrado no campo de visão da aplicação), que podem ser objetos 3D, ou provisionado de *websites*. Nenhuma das demais aplicações estudadas possuem em sua concepção a integração entre dados geográficos e sobreposição de objetos 3D. Através do *Layar*, é possível criar aplicações que integrem essas funcionalidades, pois a biblioteca possui a capacidade de criação de camadas independentes com diferentes tipos de conteúdo.

Através da biblioteca de desenvolvimento, é possível gerar uma aplicação nativa e também comercializá-la. O SDK é disponibilizado gratuitamente por 30 dias, para testes. Depois desse período, é cobrada uma taxa de licenciamento.

A ferramenta possui uma grande quantidade de documentação e também uma grande quantidade de desenvolvedores. As dúvidas são respondidas através de um fórum no site da empresa.

Sendo assim, o uso do *Layar* se justifica quando é necessário integração e uso de diversas camadas de dados, podendo ser 2D e 3D, recursos adversos de diferentes ferramentas para facilitar a criação e uso do aplicativo.

2.6.6 Comparações

Como pode ser visto na Tabela “Comparações”, dentre os *frameworks* pesquisados o que oferece mais recursos ao desenvolvimento de aplicações de Realidade Aumentada é a biblioteca *Vuforia*, que permite também rastrear imagens, objetos reais simples e marcadores comuns e sobrepor objetos tridimensionais sobre eles.

As demais aplicações são mais utilizadas pelo usuário final, como navegadores de informações no mundo real, com destaque para o *Wikitude*, que consegue pesquisar diversas fontes de dados online trazendo uma grande variedade de conteúdo.

Em nenhuma das bibliotecas estudadas observou-se a preocupação em trazer suporte multiplataforma. Isso faz com que alguns usuários sejam excluídos dessa faixa do mercado de dispositivos móveis, mesmo em que algumas plataformas que têm mostrado acentuado crescimento nos últimos anos – especialmente a *Windows Phone*.

Outro ponto de interesse refere-se ao custo. O *Wikitude* apresenta elevado custo de licenciamento. O *Layar*, como foi discutido, possibilita um período de avaliação, mas uma taxa de licenciamento é cobrada depois. Já as demais plataformas apresentam custo zero nas licenças. Apesar disso, a *Vuforia* é a única onde mais facilmente se desenvolve aplicações comerciais, pois não conta com as restrições da licença GPL V3.

No contexto das aplicações, em alguns casos, é viável o desenvolvimento em camadas para uma biblioteca já existente. O uso dessas bibliotecas, que mais parecem navegadores de RA, é viável e indicado para os casos em que não se precisa de uma infraestrutura própria para a aplicação e quando não há necessidade de um aplicativo instalado – casos em que somente uma camada (ou *layer*) de RA seja suficiente.

Outro fator que contribui o quanto a utilização de uma plataforma já consolidada de realidade aumentada auxilia o processo de aprendizado. Pode-se fazer o uso do software no modelo SaaS (software como serviço), fazendo o uso da ferramenta como infraestrutura para desenvolvimento do aplicativo como um módulo do aplicativo já existente.

A principal vantagem observada na biblioteca *Layar* manifesta-se quando se torna necessário o uso de objetos 2D e 3D no mesmo ambiente pois, dos aplicativos apresentados, apenas o *Layar* integra várias camadas web e objetos 3D no mesmo ambiente.

Tabela 1 – Comparações entre as ferramentas de autoria de Realidade Aumentada.

Biblioteca	Layar	Wikitude	Vuforia	Mixare	ARViewer
Código aberto ou gratuito?	-	✓	✓	✓	✓
Pode ser comercializado?	✓	✓	✓	-	-
Gera uma aplicação Nativa?	✓	-	✓	✓	✓
Possui recursos de Geolocalização?	✓	✓	✓	✓	✓
Comunidade de desenvolvimento?	✓	-	✓	-	-
Documentação existente?	✓	✓	✓	✓	-
Possui suporte técnico?	✓	✓	✓	-	-
Suporte a objetos 3D?	✓	-	✓	-	-
Código em Múltiplas plataformas?	-	-	-	-	-

CAPÍTULO 3

Trabalhos correlatos

3.1 Introdução

Neste Capítulo, são apresentados trabalhos relacionados a navegação baseados em geolocalização que fazem uso de diferentes tecnologias. São expostos projetos com o uso de Realidade Aumentada em diferentes áreas.

3.2 *Augmented Navigation in Outdoor Environments*

O *framework*, baseado em RA, oferece informações como imagens e descrição de um local identificado e permite a geração de itinerários para navegação baseados em Web Semântica. A aplicação integra bússola, GPS, câmera e acelerômetro.

O protótipo descrito neste trabalho apresenta uma aplicação móvel com as seguintes capacidades:

- Identificação, localização e visualização de locais de interesse próximos ao usuário;
- Procura por locais de interesse, utilizando processamento semântico baseado nas preferências e nos hábitos do usuário;
- Navegação em interface de Realidade Aumentada que combina itinerários de rotas com informações adicionais.

Foram realizados testes no centro histórico “Zocalo” no México. O aplicativo está disponível para as plataformas *iOS* e *Android*.

Um recurso interessante é a identificação de um local a partir de uma imagem real vindas do aplicativo móvel. A navegação pode ser complementada por informações de direção, usando uma simbologia específica, integrando o mundo real e o virtual.

A Figura 4 apresenta a interface que sugere possíveis locais vizinhos ao usuário, encontrados via geolocalização. À direita, pode-se ver uma barra deslizante na qual é possível



Figura 4 – *Equalizer augmented view* (MATA; CLARAMUNT, 2013).

aumentar ou diminuir o raio de distância dos locais de interesse, que interagem com o ambiente real captado pela câmera.

Este trabalho apresenta uma solução de RA para navegação outdoor que busca direcionar o usuário a um ponto de interesse. Ele se diferencia do trabalho proposto principalmente por não estar disponível para múltiplas plataformas.

3.3 *Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms*

A técnica, chamada de *Attention Funnel*, ou “funil de atenção”, propõe uma metodologia para interface de RA que orienta, de forma interativa, a atenção do usuário para qualquer objeto, pessoa ou local. Utiliza-se da dinâmica de percepção cognitiva para chamar a atenção do usuário para o local de destino. A interface pode ser usada para sinalizar objetos completamente fora de vista, incluindo os que se localizam atrás do usuário ou obstruídos por outros objetos ou paredes.

Técnicas de RA em ambientes completamente móveis possibilitam ambientes de computação onipresente (MORAN; DOURISH, 2001), oferecem a possibilidade de apoiar os usuários com uma estruturada sobreposição de grandes volumes de informação espacial tridimensional em qualquer lugar no espaço interior ou exterior, como salas de trabalho, fábricas, ruas, ou em ambientes ao ar livre abertos.

As coordenadas espaciais de objetos físicos e locais que serão aumentados usando este processo podem ser recuperada a partir de coordenadas de GPS (HOFMANN-WELLENHOF; LICHTENEGGER; COLLINS, 2004), sistemas de rastreamento (FOX-

LIN; HARRINGTON; PFEIFER, 1998), marcadores visuais, como marcadores fiduciais (KATO; BILLINGHURST, 1999) ou *tags* de rádio frequênciia (HAHNEL et al., 2004). A apresentação de informações virtuais nos objetos, tais como *labels*, sobreposições, objetos 3D adicionais, e outros dados estão integrados no ambiente físico, usando uma variedade de dispositivos de visualização que fazem as anotações virtuais parecerem ser elementos do ambiente real.

Interfaces de RA móvel apresentam desafios relacionados a interface que podem ser caracterizadas da seguinte forma: Como uma interface móvel pode gerenciar e orientar a atenção visual para locais no ambiente onde as informações críticas ou objetos estão presentes, mesmo quando eles não estão dentro do campo visual? O desafio faz parte de uma necessidade maior para a gestão da atenção do usuário (VERTEGAAL, 2002) em interfaces móveis com uma larga banda de informações.

O paradigma do funil de atenção em RA representa um exemplo de aumento cognitivo adaptado especificamente para usuários de sistemas de RA móvel navegando e trabalhando em ambientes contendo informação e objetos ricos. Os resultados experimentais da avaliação inicial mostram que o funil de atenção levou a maior consistência e menor tempo de busca e carga de trabalho mental.

As Figuras 5 e 6 apresentam o funcionamento desta ferramenta.

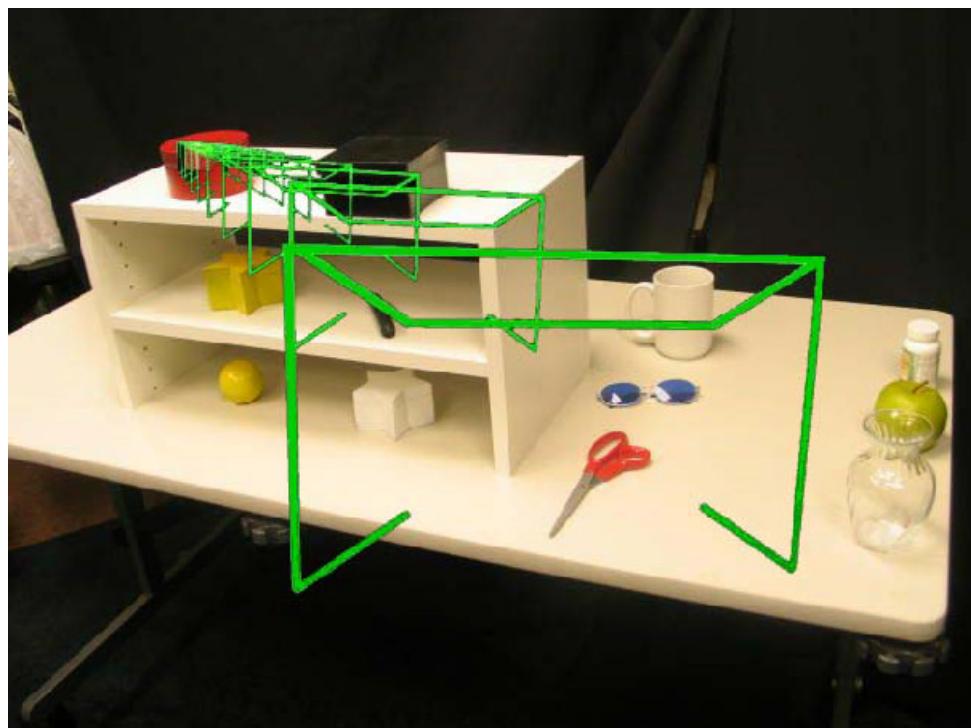


Figura 5 – Exemplo do funil de atenção focando o usuário para um objeto na prateleira: a caixa vermelha (BIOCCA et al., 2006).

Este trabalho apresenta uma solução de visualização da informação em interfaces de RA que sugere direcionar o usuário a um ponto de interesse ou objeto de interesse, mesmo



Figura 6 – Ambiente de Teste: O usuário, inserido no ambiente de teste para a tarefa de busca visual. Este ambiente consistia de um espaço de trabalho omnidirecional montado a partir de quatro mesas, cada uma com 12 objetos (6 formas primitivas e seis objetos de escritório em geral) para um total de 48 objetos de interesse (BIOCCA et al., 2006).

que este objeto não esteja visível diretamente para o usuário, direcionando o usuário de forma simplificada.

3.4 A Mobile Indoor Navigation System Interface Adapted to Vision-Based Localization

Esse trabalho apresenta uma interface que combina RV e elementos com indicadores que ajudam a se comunicar e garantir a precisão de localização em RA. Em uma avaliação com usuários, descobriu-se que a RA foi preferida em caso de localização de confiança, mas com RV as instruções de navegação são percebidas com mais precisão em caso de erros de localização e orientação. Os indicadores adicionais mostraram potencial para fazer os usuários escolherem diferentes imagens de referência para obter uma localização confiável.

Abordagens baseadas em visão para localização *indoor* utilizando dispositivos móveis não contam com grande infraestrutura e, portanto, são escaláveis e baratas. Os requisitos especiais para uma interface de navegação do usuário para um sistema baseado em visão, no entanto, não foram investigadas até o momento.

Tais interfaces móveis devem adaptar-se à precisão de geolocalização, que se baseia fortemente em diferentes imagens de referência e fatores como a postura do telefone. Se necessário, o sistema deve motivar o usuário a apontar o dispositivo para regiões distintas, visando melhorar a qualidade de localização.

Informações adicionais tais como o tempo restante, a distância ou o número de rotações para o objetivo podem ser apresentadas ao utilizador.

Uma vez que os requisitos para a navegação em ambientes fechados (*indoor*) são diferentes da navegação ao ar livre (*outdoor*), é necessária a investigação de quais partes de informação são consideradas como importante sob essas condições especiais. Um segundo desafio é a qualidade e distinção das imagens de consulta, que impactam a localização estimada. A Figura 7 apresenta o funcionamento desta ferramenta.



Figura 7 – Um *screenshot* de um dos vídeos utilizados no estudo (MÖLLER et al., 2012).

Este trabalho apresenta uma solução de RA para navegação *indoor*, que busca direcionar o usuário a um ponto de interesse em ambiente fechado, a interface que apresenta as informações ao usuário é semelhante ao trabalho proposto, se diferenciando por trabalhar em ambientes fechados e por não estar disponível para múltiplas plataformas.

3.5 *VisitAR: a mobile application for tourism using AR*

Essa aplicação, apresentada na Figura 8 é resultado da combinação de duas interfaces, o *Glue channel* (GC) e o canal LBC baseado em localização. O GC permite a visualização de um mapa de RA interativo, destinado a ajudar turistas no planejamento dos passeios. Exibe-se um mapa tradicional, sobreposto com camadas virtuais, textos e modelos em 3D das principais atrações. Esses pontos de interesse são apresentados de uma forma que permite ao usuário personalizar livremente o número e o tipo de atrações exibidas no mapa. Além disso, informações sobre atrações individuais são fornecidas.



Figura 8 – *Screenshots do VisitAR application* (MICHELE; MICHELE; FABIO, 2013).

O passeio guiado foi realizado utilizando o LBC, que faz uso de GPS e da bússola para o rastreamento do dispositivo/usuário. A interface do aplicativo desenvolvida está em uma fase preliminar de estudos, a fim de criar uma forma simples e intuitiva para obter informações da cidade. Particularmente um ponto importante do aplicativo é a personalização dos pontos de interesse, mostrando, usando diferentes categorias como monumentos, restaurantes, lojas e muito mais.

O principal objetivo deste projeto é desenvolver uma aplicação/guia para o turismo. O estudo de caso é sobre a cidade de Oulu, escolhida graças à disponibilidade de um modelo 3D da cidade, usado para o módulo GC, e sua grande cobertura de rede *Wi-Fi*. Para o módulo LBC, levantou-se vários pontos de interesse como amostra, a fim de criar uma base de pontos de interesse para construir o aplicativo de teste.

Foram armazenadas informações sobre 30 pontos de interesse (em torno de quatro para cada categoria), incluindo fotos, vídeos e descrições. Foram adicionados também alguns modelos 3D de monumentos da cidade, o mesmo utilizado no GC. Desta forma, utilizando o modo ao vivo pela LBC, os usuários podem ver os monumentos em 3D através da câmera.

Atualmente, os monumentos indicados são os mesmos existentes no mundo real, mas no futuro pretende-se tornar possível a colocação de modelos 3D antigos ou novos criados. Desta forma, será possível adicionar facilmente modelos 3D, como conteúdo da aplicação.

Cada ponto de interesse contém vários dados: informações relacionadas ao GPS (longitude, latitude, altitude), título, descrição, e, eventualmente, imagens ligadas, vídeos e *links*.

Todas as informações dos pontos de interesse são armazenadas em um banco de dados e são recuperados usando um *script* que busca os dados do servidor. As informações são enviadas ao dispositivo sempre que este faz uma requisição.

Este trabalho apresenta uma solução de RA para navegação *outdoor*, que busca direcionar o usuário baseado em pontos de interesse, a interface que apresenta as informações ao usuário, com opções de filtros, semelhante ao trabalho proposto, se diferenciando por não estar disponível para múltiplas plataformas.

3.6 *Building Mobile AR Applications Using the Outdoor AR Library*

O *HITLab*, da Nova Zelândia, desenvolveu um *framework* de RA Móvel, que é uma plataforma de desenvolvimento construída e utilizada na construção de aplicações de RA móvel e protótipos para a pesquisa de tecnologia de interface humana. Ele consiste em várias bibliotecas e ferramentas integradas que auxiliam os desenvolvedores na criação de sua própria experiência baseada em RA móvel.

A Realidade Aumentada para navegação Outdoor combina localização em escala global (ou geolocalização), tecnologia de rastreamento e tecnologia de computação gráfica para mostrar informações virtuais sobrepostas em uma visão de câmera ao vivo, de forma que objetos virtuais aparecem registrados em locais do mundo real.

Além de fornecer informações sobre os pontos de interesse (POI) no mundo real, os sistemas de RA ao ar livre tornaram-se capazes de proporcionar experiências mais interativas, como um sistema guia melhorado com RA *outdoor*.

Para desenvolver a biblioteca para RA *outdoor*, foi utilizado um *framework* de desenvolvimento de software, implementada para a plataforma *Android* (versão 2.2 ou superior) baseado no *Android* SDK, apresentado na Figura 9

3.7 *Mobile AR applications and services for use in the City of Basel*

Basel é uma cidade dotada de um guia turístico baseado em RA, acessível através do navegador *Layar*. Usuários podem obter informações sobre a cidade de Basel, em especial sobre seus museus, restaurantes, hotéis, shopping centers, eventos, ferramentas de software na Internet, bem como as habilidades dos desenvolvedores de RA móvel em todo o mundo.

O objetivo deste projeto é estabelecer um *framework* no qual profissionais criativos possam ver o que foi feito em Cidades Inteligentes, com conjuntos de dados 3D e para mostrar a primeira geração de serviços que estão disponíveis ao público com *smartphones*



Figura 9 – Protótipo de aplicação baseado no *framework* do *HITLab* (LEE; BIL-LINGHURST, 2013).

durante a conferência ISMAR. Este projeto coordena recursos na cidade de Basel, e internacionalmente com a finalidade de implementar serviços de RA móvel no mundo real. A Figura 10

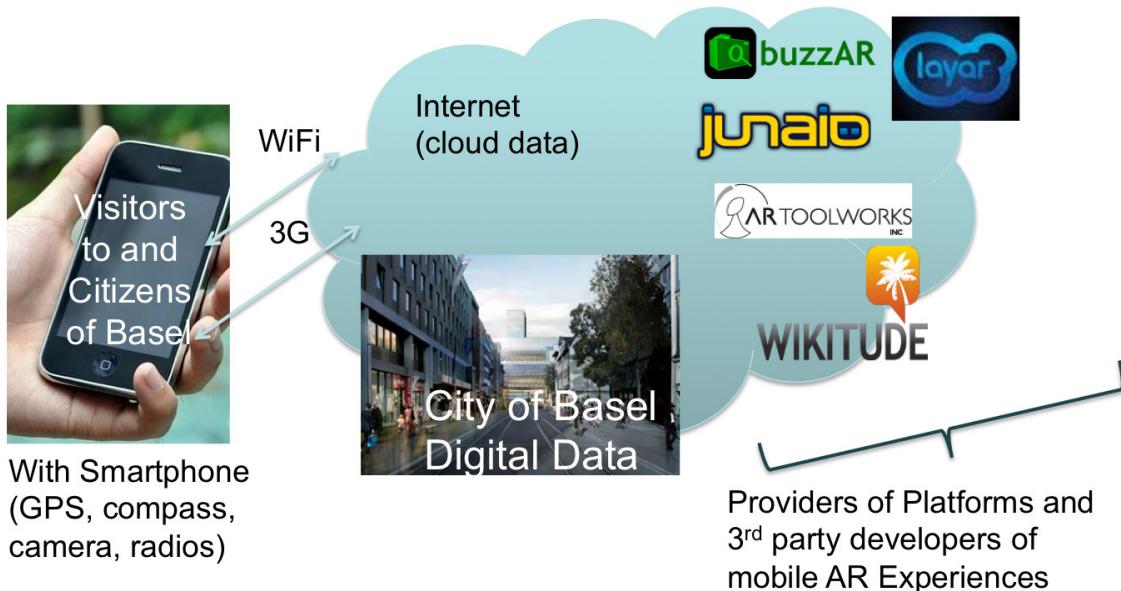


Figura 10 – Esquema básico do Projeto *Basel* (PEREY; MIYASHITA, 2011).

Em sua pesquisa, os autores listam os principais problemas encontrados na implementação de uma aplicação de RA para turismo. Como por exemplo, o custo elevado em uso de dados, poucas redes *wi-fi* disponíveis e também pagamento *roaming* de dados para estrangeiros. O maior obstáculo encontrado pelos autores diz respeito a falta de interoperabilidade entre plataformas móveis.

Apesar das dificuldades, eles argumentam que empregar RA no turismo pode ajudar os turistas no planejamento de viagem, informações úteis, etc. Aumentado seus conhecimentos a respeito a um lugar ou atração turística.

3.8 Trabalhos Analisados

Os trabalhos apresentados neste capítulo exibem diversas aplicações de ambientes virtuais e aumentados aplicados a dispositivos móveis com geolocalização. Para que pudessem ser comparados, esses trabalhos foram analisados sob diferentes ópticas, que podem ser resumida nas seguintes questões:

- ❑ Navegação outdoor:
 - Trabalha com coordenadas reais?
 - Obtém localização do usuário?
 - Calcula distância do usuário a um determinado ponto de interesse?
 - Direciona usuário para o ponto de interesse?
- ❑ Camadas:
 - Apresenta uma camada com informações de pontos de interesse?
- ❑ Realidade aumentada:
 - Exibe o mundo real sobreposto através de uma superfície de RA?
- ❑ Desenvolvimento multiplataforma
 - Ferramenta baseada em linguagens multiplataforma?
 - Desenvolvimento várias versões de um mesmo aplicativo para diferentes plataformas?
 - Exige conhecimentos em diferentes plataformas e tecnologias, gerando maior dificuldade e maior demanda por tempo durante o desenvolvimento?

Esses quatro itens, aplicados aos trabalhos vistos nesse Capítulo e o resultado pode ser visto na Tabela 2. O principal ponto de destaque é no que se refere ao suporte multiplataforma, não oferecido por nenhum deles. Esse é um dos aspectos mais importantes do presente trabalho.

Tabela 2 – Comparações entre os trabalhos correlatos analisados nas Seções 3.2, 3.3, 3.4, 3.5, 3.6 e 3.7

Seção	3.2	3.3	3.4	3.5	3.6	3.7
Geolocalização?	✓	✓	-	✓	✓	✓
Camadas?	✓	✓	✓	✓	✓	✓
RA?	✓	✓	✓	✓	✓	✓
Multiplataforma?	-	NA	-	-	-	-

3.9 Conclusão

A partir dos trabalhos apresentados, pode-se concluir que até o presente momento diferentes tipos de aplicações de RA Móvel voltadas para navegação dos usuários foram concebidos. No entanto, nenhum deles tem em conta a experiência de um mesmo aplicativos em diferentes plataformas, a partir do momento em que chegam no processo de desenvolvimento esses aplicativos estão disponíveis para apenas uma ou poucas plataformas. Por esta razão, pensamos em um aplicativo que leva em conta a diversidade de Sistemas Operacionais Móveis e nós fornecemos um protótipo disponível para o maior número de dispositivos possíveis utilizando uma ferramenta que possibilita o desenvolvimento de uma forma simples e intuitiva.

Outra lição aprendida pelos trabalhos apresentados, considera situações críticas para a utilização de RA em aplicações de geolocalização, tais como a disponibilidade de conexão *Wi-Fi*, o custo de conexões de dados móveis e a interoperabilidade entre plataformas móveis.

CAPÍTULO 4

Especificação do Sistema Proposto

4.1 Introdução

Este capítulo apresenta as especificações do sistema de acordo com as normas comuns de Engenharia de Software. São apresentados os requisitos funcionais e não-funcionais do sistema e seus casos de uso detalhados.

4.2 Requisitos Funcionais

Segundo Sommerville et al. (2008), os requisitos funcionais para um sistema indica o que este deve fazer, ou seja, como reage ao receber entradas específicas e em determinadas situações. Os principais requisitos funcionais de um sistema de Geolocalização utilizando RA Móvel envolvem a correta aquisição dos dados, o processamento dos mesmos e a apresentação da simplificação dos dados obtidos.

4.2.1 Aquisição de dados

O sistema de geolocalização desenvolvido neste trabalho necessita de apenas duas informações por parte do usuário: sua localização atual e seu destino. De fato, a localização atual é obtida automaticamente pelo sistema GPS do aparelho, a localização de destino é obtida através das coordenadas do ponto de interesse determinado.

Para que a coleta de tais dados seja feita de maneira correta, é preciso levar em consideração as características físicas do dispositivo móvel, como a existência de GPS, bem como se o mesmo está habilitado para uso.

Para que o aplicativo possa ser utilizado em diferentes dispositivos, é importante estabelecer o formato dos dados coletados a ser utilizada em qualquer caso de uso.

4.2.2 Processamento de informações em tempo real

Para que o usuário tenha como resposta a distância e direção no momento em tempo real, é necessária a geração das informações durante o processo de coleta de dados. Para isso, são utilizadas duas fórmulas distintas: uma para cálculo da distância e outra para cálculo da direção a seguir.

4.2.3 Criação do ambiente real aumentado

Esta etapa do processo requer bastante atenção. A sobreposição do mundo real, através do objeto virtual, necessita de uma série de parâmetros que devem ser bem ajustados/calculados. Entre eles, destacam-se as características do ambiente aumentado:

- ❑ Apresentação da informação de distância do destino;
- ❑ Direcionamento do destino;
- ❑ Apresentação do mundo real sem que as informações virtuais poluam a visão do usuário, para que a experiência seja realizada da melhor forma possível.

4.3 Requisitos não-funcionais

Requisitos não-funcionais são aqueles que definem restrições sobre os serviços ou as funções que o sistema oferece (SOMMERRVILLE et al., 2008). A seguir, serão discutidos os requisitos referentes à implementação, portabilidade e eficiência.

4.3.1 Requisitos de Implementação

O sistema deverá ser desenvolvido utilizando o *framework PhoneGap*, utilizando as linguagens HTML, CSS e *Javascript*.

4.3.2 Requisitos de Portabilidade

O aplicativo deve ser compatível a dispositivos com o sistema *Android* com versão 4.0 ou superior, *Windows Phone 7* ou superior e *iOS 4* ou superior. Embora o código fonte do aplicativo seja exatamente o mesmo para todas os sistemas operacionais, é necessário gerar complicações compatíveis com cada um. Para o *Android*, por exemplo, é gerado um arquivo com extensão *.apk*.

4.3.3 Requisitos de Eficiência

Para que o usuário tenha melhor experiência de utilização, as informações de distância e direção devem ser atualizadas a cada dois segundos. O GPS do aparelho deve estar habilitado no momento de inicialização do aplicativo.

4.4 Caso de Uso

Nesta seção, o ator, chamado “usuário”, é descrito e suas principais atribuições são apresentadas. O usuário é quem manipula o dispositivo móvel e interagem com a aplicação. Sua descrição está resumida na Tabela 3.

Tabela 3 – Ator do sistema.

Nome	Descrição	Atribuições
Usuário	Utilizador do sistema	Iniciar a aplicação de Realidade Aumentada. Visualizar o ambiente aumentado, apresentado na tela do telefone, exibindo o mundo real, capturado através da câmera e sobreposto com informações referentes a distância e direção do ponto de interesse, mostrada por uma seta. Seguir a direção apontada pela seta. Fechar a aplicação.

Na Figura 11, é apresentado o principal caso de uso da aplicação. Sua descrição é dada a seguir, distribuída nas Tabelas 4, 5, 6, 7, 8 e 9.

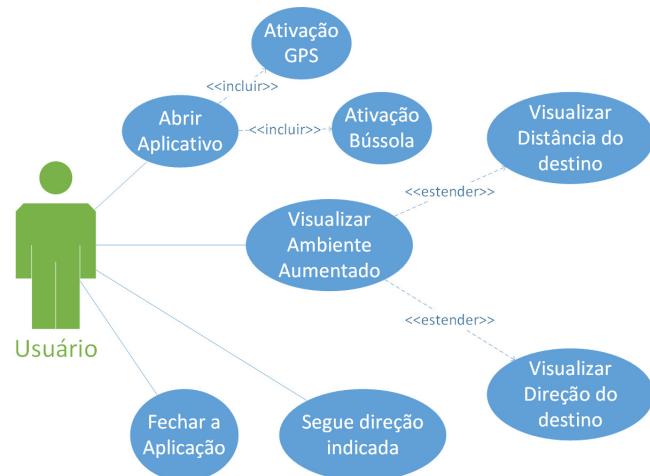


Figura 11 – Caso de uso do sistema.

Tabela 4 – Descrição do caso de uso “Iniciar a aplicação”

Caso de uso	Iniciar a aplicação
Ator Principal	Usuário
Descrição	Iniciar a execução do aplicativo de Realidade Aumentada. Ao iniciar, a aplicação irá ativar a captura dos dados de localização e a bússola para captura da direção do norte magnético em relação a atual localização.
Pré-Condição	Estar habilitado o fornecimento das informações do GPS e localização nas configurações do dispositivo. Bússola estar calibrada.
Fluxo Normal	<ol style="list-style-type: none"> 1. Executar a aplicação.
Fluxos Excepcionais	-
Pós-Condição	<ul style="list-style-type: none"> <input type="checkbox"/> GPS: Ativo <input type="checkbox"/> Bússola: Ativa

Tabela 5 – Descrição do caso de uso “Visualizar Ambiente Aumentado”

Caso de uso	Visualizar Ambiente Aumentado
Ator Principal	Usuário
Descrição	Uma vez exibido o ambiente no dispositivo, visualizá-lo em Realidade Aumentada.
Pré-Condição	Estar com o sistema aberto, ativado o GPS e Bússola e gerado o ambiente virtual.
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a aplicação; 2. Posicionar o dispositivo em frente o rosto; 3. Visualizar ambiente aumentado.
Fluxos Excepcionais	-
Pós-Condição	Reprodução do ambiente em RA

Tabela 6 – Descrição do caso de uso “Visualizar Distância do destino”

Caso de uso	Visualizar Distância do destino
Ator Principal	Usuário
Descrição	Uma vez exibido o ambiente de RA, visualizar a informação referente a distância em relação ao ponto de interesse.
Pré-Condição	Estar com o sistema aberto, ativado o GPS e Bússola e gerado o ambiente virtual.
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a aplicação; 2. Posicionar o dispositivo em frente o rosto; 3. Visualizar ambiente aumentado; 4. Visualizar o campo que apresenta a distância do destino.
Fluxos Expcionais	-
Pós-Condição	-

Tabela 7 – Descrição do caso de uso “Visualizar Direção do destino”

Caso de uso	Visualizar Direção do destino
Ator Principal	Usuário
Descrição	Uma vez exibido o ambiente de RA, visualizar a seta referente a direção a do ponto de interesse.
Pré-Condição	Estar com o sistema aberto, ativado o GPS e Bússola e gerado o ambiente virtual.
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a aplicação; 2. Posicionar o dispositivo em frente o rosto; 3. Visualizar ambiente aumentado; 4. Visualizar a seta que direciona para o destino.
Fluxos Expcionais	-
Pós-Condição	-

Tabela 8 – Descrição do caso de uso “Seguir direção indicada”

Caso de uso	Seguir direção indicada
Ator Principal	Usuário
Descrição	Uma vez conhecidas as informações de distância e direção, seguir a direção indicada.
Pré-Condição	Uma vez exibido o ambiente de RA, visualizar a informações para se chegar ao destino.
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a aplicação; 2. Posicionar o dispositivo em frente o rosto; 3. Visualizar ambiente aumentado; 4. Visualizar a distância do ponto de interesse; 5. Visualizar a seta que direciona para o destino; 6. Seguir em direção ao destino.
Fluxos Expcionais	-
Pós-Condição	Chegar ao local desejado.

Tabela 9 – Descrição do caso de uso “Fechar a Aplicação”

Caso de uso	Fechar a Aplicação
Ator Principal	Usuário
Descrição	Fechar a Aplicação
Pré-Condição	
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a aplicação; 2. Visualizar ambiente de RA; 3. Fechar a aplicação utilizando o botão “voltar” por duas vezes.
Fluxos Expcionais	-
Pós-Condição	Aplicação finalizada.

4.5 Diagrama de sequência

O fluxo de dados orientado pelas funções a serem executadas é descrito na Figura 12. Essas transformações funcionais convertem as entradas do software (Coordenadas do GPS e sentido da Bússola) em saídas (Interface de RA)

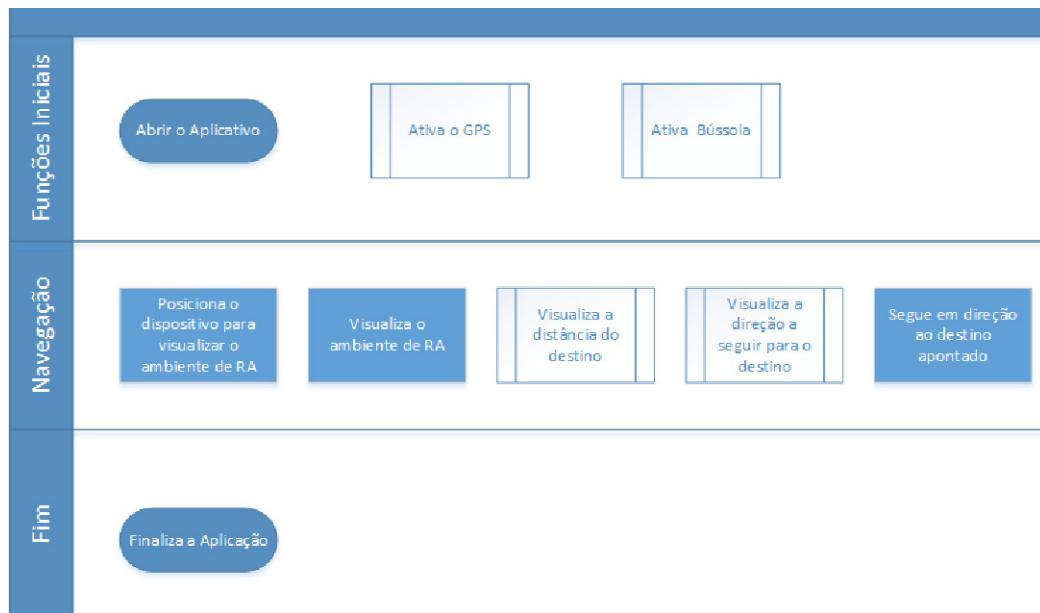


Figura 12 – Diagrama de sequência de utilizações do sistema.

CAPÍTULO 5

Detalhes da Implementação

5.1 Introdução

Este capítulo apresenta uma visão geral das funcionalidades da aplicação desenvolvida, como prova de conceito, para utilização do *PhoneGap* e como o *framework* foi utilizado para o seu desenvolvimento.

O objetivo da aplicação é apresentar a geolocalização dentro de um Campus em uma Universidade. Esta aplicação utiliza a tecnologia Realidade Aumentada acrescentando ao mundo real, a informação virtual da seta indicando o sentido do caminho para localizar a direção de destino em relação ao ponto onde o usuário está.

5.2 Obter a Posição do Usuário

O recurso de Geolocalização fornece informações sobre a localização do dispositivo, tais como latitude e longitude. As fontes comuns de informação de localização incluem Sistema de Posicionamento Global (GPS) e localização inferida a partir de sinais de rede, como endereço IP, RFID, *WiFi* e *Bluetooth* endereços MAC, e IDs de GSM/CDMA celulares. Esta API é baseada na W3C (STANDARDS, 2013) *Geolocation API Specification*.

A definição encontrada na documentação referente a classe *geolocation* é a seguinte: “O objeto *geolocation* fornece acesso para o sensor GPS do dispositivo”.

Foi utilizado o método *geolocation.getCurrentPosition()* para pegar a posição atual do dispositivo, essa é uma função assíncrona, ela retorna a posição atual do dispositivo ao parâmetro *geolocationSuccess*. Foi atribuído a uma variável esse valor para cálculo da distância entre o ponto no qual o usuário está e o ponto de destino do mesmo.

Como o usuário estará em movimento, o método *geolocation.watchPosition*, que também é uma função assíncrona irá retorna a posição atual do dispositivo quando qualquer mudança de posição foi detectada.

5.3 Carregar a camada com informações de pontos de interesse

Após obter os dados da posição em que o dispositivo está, a segunda etapa foi obter os pontos onde o usuário deseja visitar, onde deverá ser calculada a distância entre o ponto inicial e final, que foram definidos em uma lista de latitude e longitude em um arquivo no *Javascript*, chamado locais.js, que permite serem trabalhados dentro da aplicação.

A ideia inicial era exibir um mapa, com coordenadas reais de cada ponto de interesse, mas como trata-se da distância em linha reta entre os pontos, dispensando a necessidade de um mapa, bastando fazer um cálculo matemático segundo o teorema de Pitagoras, para isso considera-se que os dois pontos estejam no mesmo plano de fundo, visto que a aplicação é feita para uma mesma cidade, o que faz com que seja mínima a diferença caso fosse considerado também a altitude, o que pode ser incorporado em um versão posterior.

Essa camada locais.js contém as informações de todos os pontos de interesse criados para a realização dos testes.

5.4 Cálculo da Distância entre Pontos

Tendo em mãos a localização do usuário e os pontos de interesse, a terceira etapa foi realizar o cálculo da distância entre o ponto em que o dispositivo está e um determinado ponto cadastrado.

Para encontrar o cálculo da distância entre dois pontos, considerando que a Terra é uma esfera perfeita, o cálculo é dado por raiz quadrada de $x^2 + y^2$, onde x é latitudePonto1-latitudePonto2 e y é longitudePonto1-longitudePonto2.

Foi criada o arquivo calculos.js que contém uma classe para o cálculo da distância que recebe os parâmetros latitudePonto1, longitudePonto1, latitudePonto2, longitudePonto2 e retorna a variável distancia. Desta forma, foi encontrada a distância entre o ponto inicial e o ponto que se deseja chegar.

5.5 Cálculo da *azimute*

Para obter a direção de destino em relação ao ponto onde o usuário está, deve ser feito o cálculo da *azimute*, ou seja, o ângulo entre os dois pontos geográficos.

O termo *azimute*, de acordo com Gaspar (2005), é de origem árabe (as-sumut) e significa “caminho ou direção”. Por definição, o azimute de um objeto, segundo Gaspar (2005), é o ângulo numa posição determinada, entre um arco de círculo máximo e o meridiano que passa nessa posição ou como definiu ELMIRO (2012), “é a forma mais usada para indicar uma direção geográfica. O *azimute* é o ângulo formado entre a direção Norte (meridiano ou *azimute* zero) e uma direção terrestre considerada”. O *azimute* que

vamos obter no aplicativo é o cartográfico, que é medido a partir do Norte Cartográfico (direção das linhas verticais das quadrículas cartográficas).

Para obter a *azimute*, no arquivo calculos.js foi criado a classe *azimute*, que a partir da diferença entre as latitudes e as longitudes dos dois pontos, é realizado o seguinte cálculo $azimute = (Math.PI * 0.5) - Math.atan(diferencaLatitudinal / diferencaLongitudinal);$.

Desta forma, foi calculada a azimute de cada ponto.

5.6 Bússola e *azimute*

Com as coordenadas do GPS e também os campos calculados de distância e *azimute*, também foi necessário o valor da bússola para que todas as informações para criação da rosa dos ventos de direção estejam presentes na aplicação.

Para obter o ângulo da bússola, no caso do dispositivo móvel o que se encontra é um Magnetômetro, que utiliza o mesmo princípio da bússola, de magnetismo em relação ao polo magnético da Terra, e faz parte do hardware de alguns modelos de smartphones.

O *azimute* é retornado através do objeto *compass.getCurrentHeading*, que informa o ângulo, que é mede o cabeçalho em graus, entre 0 e 359,99. O método *compass.watchHeading* recebe o valor atual da bússola do dispositivo em um intervalo regular. Cada vez que esse valor é alterado, a função *headingSuccess* de retorno é executada.

5.7 Rosa dos ventos e distância do destino

Para fornecer as informações de distância e direção, foram criados dois componentes na interface, criada em HTML5, um label e uma imagem. Em Javascript foi feita a chamada ao método de cálculo da distância e atribuído ao elemento HTML *distancia* o valor retornado da função *document.getElementById('distancia').innerHTML = distancia(coords.latitudeOrigem, coords.longitudeOrigem, coords.latitudeDestino, coords.longitudeDestino);*. Desta forma, é exibida na tela a distância do ponto de interesse.

Além do cálculo da azimute, para a seta de direção deverá ser subtraído o valor da azimute, que é o ângulo do ponto onde eu estou em relação ao ponto onde quero chegar, e o ângulo da bússola. A função *angulo = (GetAzimuteGraus(coords.latitudeOrigem, coords.latitudeDestino, coords.longitudeOrigem, coords.longitudeDestino) - valorBussola);* realiza esse cálculo e retorna o valor que deve ser atribuído ao ângulo da imagem, a função *img.stop().animate(transform: "r"+ angulo , 1, <>");* direciona a direção que a imagem deve estar apontando.

Um cálculo deve ser feito para o caso da subtração do ângulo da bússola e a azimute entre os dois pontos seja um resultado menor que zero, é necessário acertar para um valor entre 360 graus, para esses casos, é somado a variável *angulo* o 360.

Na tela inicial da aplicação são apresentadas as informações de coordenada, distância e seta, conforme apresentado na Figura 13.



Figura 13 – Tela inicial da aplicação apresentando o status do GPS e da bússola.

Ao abrir a aplicação, será exibida uma tela de visualização da seta direcional. O usuário deverá então utilizar a seta para se guiar em direção ao ponto de destino. De acordo com a movimentação do usuário a direção e distância é alterada em relação ao novo ponto, conforme apresentado nas Figuras 14 e 15.



Figura 14 – Informações de distância , do GPS e seta apontando para destino atualizadas em tempo real.

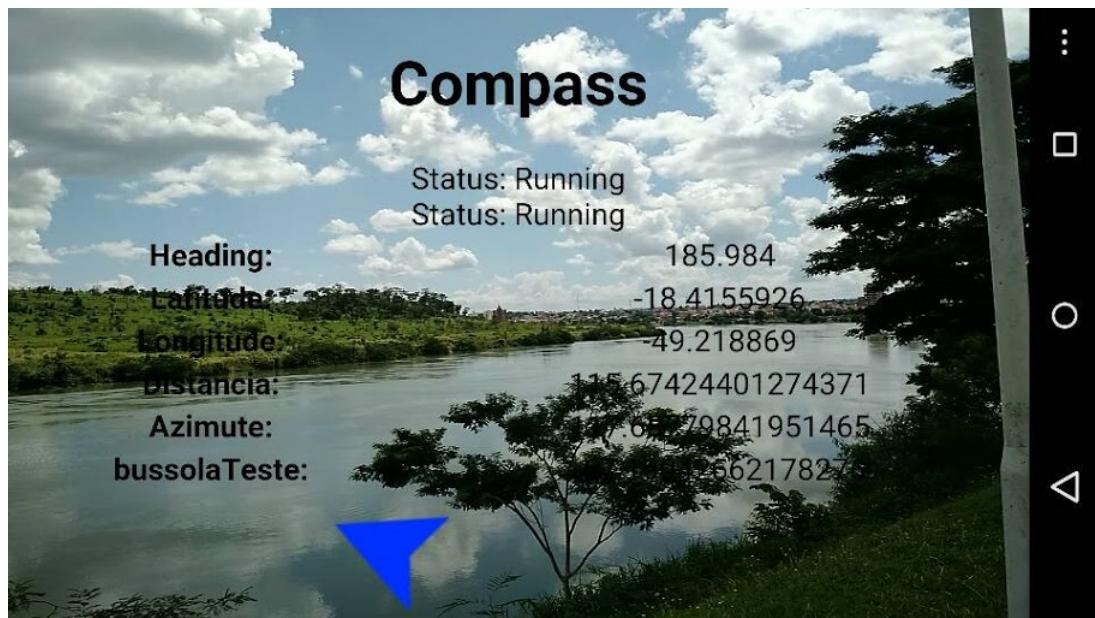


Figura 15 – App em funcionamento. Nota-se que as informações são sobrepostas às imagens do mundo real.

CAPÍTULO 6

Discussão dos resultados

6.1 Introdução

No Capítulo 2, os conceitos-chave para o entendimento da RA Móvel foram apresentadas, juntamente com a visão geral de várias bibliotecas para este fim. Com base nestes conceitos, foi possível, no Capítulo 4, especificar um aplicativo para dispositivos móveis para geolocalização baseado em RA para ser executado em múltiplas plataformas e cujos detalhes de implementação foram apresentados no Capítulo 5.

No presente Capítulo, uma análise comparativa entre a aplicação desenvolvida e os trabalhos correlatos discutidos no Capítulo 3 é realizada.

6.2 Analise Comparativa

Por se tratar de uma prova de conceito, o *app* desenvolvido não foi publicado em lojas de aplicativos, sendo apenas instalado localmente em *smartphones* com diferentes sistemas operacionais onde os testes foram realizados.

A facilidade de navegação interação por parte do usuário foram as principais características que nortearam o desenvolvimento da interface gráfica do aplicativo. O usuário é orientado a chegar ao ponto de interesse, cuja direção é visualizada através de uma seta, e a distância do ponto (em metros) é também apresentada, conforme foi apresentado nas Figuras 13 e 14 do Capítulo 5.

Com as análises dos trabalhos correlatos, realizadas na Seção 3.8 e sintetizadas na Tabela 2, foi possível perceber que o suporte ao desenvolvimento de aplicações multiplataforma é o principal desafio enfrentado no desenvolvimento de RA Móvel. O *app* criado no presente trabalho, porém, conseguiu, graças às possibilidades criadas pelo *framework PhoneGap*, executar (com base no mesmo código fonte) em diferentes sistemas operacionais.

Conclusão

Considerações finais

Este trabalho proporcionou a criação de um aplicativo para computação móvel que permite ao usuário fazer a navegação em direção a um ponto de interesse, utilizando uma linguagem multiplataforma para seu desenvolvimento. Do ponto de vista científico, as principais contribuições deste trabalho são:

- ❑ Avaliação das tecnologias para desenvolvimento de Realidade Aumentada Móvel;
- ❑ Análise de trabalhos voltados para navegação utilizando RA e suas interfaces;
- ❑ Proposta de algoritmos eficientes de desenvolvimento voltado para múltiplas plataformas, sem necessidade de grande conhecimento em linguagens direcionadas a plataformas específicas;
- ❑ Geração de um aplicativo de Realidade Aumentada que pode ser utilizado no auxílio a navegação em ambientes externos.

Trabalhos futuros

Além do aprimoramento do sistema desenvolvido nesta dissertação, a transformação deste em uma ferramenta de autoria figura como trabalho futuro. Para tanto, serão necessários os seguintes requisitos:

- ❑ Aprimoramento do sistema, a fim de fazer o uso de componentes 3D.
- ❑ Determinar um padrão para que o desenvolvimento que inclua componentes 3D seja compatível com RA Móvel multiplataforma;
- ❑ Utilizar *canvas*, *SVG* e *WebGL* para renderização de objetos 3D, que são recursos presentes no padrão *W3C* para gráficos.

- ❑ Permitir aos usuários criarem aplicações através de uma ferramenta de autoria de RA para navegação outdoor de forma simples.
- ❑ Fazer o uso de geolocalização, componentes 3D e RA a partir do *framework* integrados a uma ferramenta de autoria, visando a criação de aplicativos de RA de forma semi-automática utilizando código já existente e informações realizada através de parâmetros inseridos pelo usuário;
- ❑ Criar interface para geração de ambientes de forma simplificada, onde o usuário cadastra pontos de interesse e seleciona quais recursos serão apresentados e seja gerado um aplicativo de RA que apresente tais informações, não necessitando de grandes conhecimentos em programação;
- ❑ Habilitar a função de personalização por perfil de usuário, possibilitando a apresentação das informações que realmente são de interesse do usuário.

Referências

- APPLE. **Apple**. 2014. Disponível em: <<http://www.apple.com>>.
- AZUMA, R. et al. Recent advances in augmented reality. **IEEE Comput. Graph. Appl.**, 2001. IEEE Computer Society Press, Los Alamitos, CA, USA, v. 21, n. 6, p. 34–47, nov. 2001. ISSN 0272-1716. Disponível em: <<http://dx.doi.org/10.1109/38.963459>>.
- BIOCCA, F. et al. Attention funnel: Omnidirectional 3d cursor for mobile augmented reality platforms. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2006. (CHI '06), p. 1115–1122. ISBN 1-59593-372-7. Disponível em: <<http://doi.acm.org/10.1145/1124772.1124939>>.
- BRIGHT, P. **Windows Phone 7 nas empresas: não só boas notícias**. 2010. Disponível em: <<http://arstechnica.com/microsoft/news/201in-the-enterprise-not-all-good-news.ars>>.
- CARDOSO, A. et al. Desenvolvimento de gestos personalizados para criação e navegação em ambientes de realidade virtual e aumentada associada a dispositivos móveis. In: **XIV Simpósio de Realidade Virtual e Aumentada**. [S.l.: s.n.], 2012.
- CHANG, W.; TAN, Q. Augmented reality system design and scenario study for location-based adaptive mobile learning. In: **CSE**. [S.l.]: IEEE, 2010. p. 20–27. ISBN 978-1-4244-9591-7.
- CONSULARO, L. A.; COELHO, R. C.; JR, N. C. Rastreamento óptico para sistemas de realidade virtual e aumentada. **Realidade Virtual e Aumentada: Conceitos, Projeto e Aplicações**, 2007. p. 22, 2007.
- CORRAL, L. et al. Evolution of mobile software development from platform-specific to web-based multiplatform paradigm. In: **Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software**. New York, NY, USA: ACM, 2011. (ONWARD '11), p. 181–183. ISBN 978-1-4503-0941-7. Disponível em: <<http://doi.acm.org/10.1145/2048237.2157457>>.
- COSTA, N. P. O.; FILHO, N. F. D. Análise e avaliação funcional de sistemas operacionais móveis: vantagens e desvantagens. **Revista de Sistemas e Computação – RSC**, 2013. 2013.

- ELMIRO, M. A. T. **Curso de fundamentos de cartografia, matemática e geodésica**. UFMG, MG, Brasil, 2012. Disponível em: <<http://xa.yimg.com/kq/groups/17314041/600726175/name/Cartografia-Matem%C3%A1tica-Geodesica.pdf>>.
- FLORINS, M.; VANDERDONCKT, J. Graceful degradation of user interfaces as a design method for multiplatform systems. In: **Proceedings of the 9th International Conference on Intelligent User Interfaces**. New York, NY, USA: ACM, 2004. (IUI '04), p. 140–147. ISBN 1-58113-815-6. Disponível em: <<http://doi.acm.org/10.1145/964442.964469>>.
- FOXLIN, E.; HARRINGTON, M.; PFEIFER, G. Constellation: A wide-range wireless motion-tracking system for augmented reality and virtual set applications. In: **Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques**. New York, NY, USA: ACM, 1998. (SIGGRAPH '98), p. 371–378. ISBN 0-89791-999-8. Disponível em: <<http://doi.acm.org/10.1145/280814.280937>>.
- GARTNER. **Gartner Recommends a Hybrid Approach for Business-to-Employee Mobile Apps**. 2013. Disponível em: <<http://www.gartner.com/newsroom/id/2429815>>.
- GASPAR, J. **Cartas e projeções cartográficas**. Lidel, 2005. ISBN 9789727573714. Disponível em: <<http://books.google.com.br/books?id=FCktNQAACAAJ>>.
- HAHNEL, D. et al. Mapping and localization with rfid technology. In: **Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on**. [S.l.: s.n.], 2004. v. 1, p. 1015–1020 Vol.1. ISSN 1050-4729.
- HOFMANN-WELLENHOF, B.; LICHTENEGGER, H.; COLLINS, J. **Global Positioning System: theory and practice**. [S.l.]: Springer-Verlag, 2004. ISBN 9783211825914.
- INC, G. **2012 Press Releases**. 2013. Disponível em: <<https://investor.google.com/releases/2012/index.html>>.
- ISAACSON, W. **Steve Jobs**. Simon & Schuster, 2011. ISBN 9781451648553. Disponível em: <<http://books.google.com.br/books?id=I6R8MXStPXgC>>.
- JUNIOR, G. de P. S. et al. Realidade aumentada móvel. In: CARDOSO, A. et al. (Ed.). **tendencias e técnicas em realidade virtual e aumentada**. Porto Alegre, RS, Brasil: Sociedade Brasileira de Computação – SBC, 2013. p. 60–74.
- KATO, H.; BILLINGHURST, M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In: **Proceedings of the 2Nd IEEE and ACM International Workshop on Augmented Reality**. Washington, DC, USA: IEEE Computer Society, 1999. (IWAR '99), p. 85–. ISBN 0-7695-0359-4. Disponível em: <<http://dl.acm.org/citation.cfm?id=857202.858134>>.
- KIRNER, C.; KIRNER, T. Realidade virtual e aumentada: Aplicações e tendências. realidade virtual e aumentada: Aplicações e tendências. **Evolução e Tendências da Realidade Virtual e da Realidade Aumentada**, 2011. SBC, Porto Alegre, RS, Brasil, v. 1, n. 6, p. 8–23, 2011.

- LECHETA, R. R. **Google Android**. Novatec, 2010. Disponível em: <<http://www.in2ar.com>>.
- LEE, G. A.; BILLINGHURST, M. Building mobile ar applications using the outdoor ar library. In: **SIGGRAPH Asia 2013 Symposium on Mobile Graphics and Interactive Applications**. New York, NY, USA: ACM, 2013. (SA '13), p. 81:1–81:1. ISBN 978-1-4503-2633-9. Disponível em: <<http://doi.acm.org/10.1145/2543651.2543662>>.
- MADDEN, L. **Professional Augmented Reality Browsers for Smartphones: Programming for Juniaio, Layar and Wikitude**. 1st. ed. [S.l.]: Wiley Publishing, 2011. ISBN 1119992818, 9781119992813.
- MATA, F.; CLARAMUNT, C. Augmented navigation in outdoor environments. In: **Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems**. New York, NY, USA: ACM, 2013. (SIGSPATIAL'13), p. 524–527. ISBN 978-1-4503-2521-9. Disponível em: <<http://doi.acm.org/10.1145/2525314.2525319>>.
- MICHELE, G.; MICHELE, D. D.; FABIO, S. Visitar: A mobile application for tourism using ar. In: **SIGGRAPH Asia 2013 Symposium on Mobile Graphics and Interactive Applications**. New York, NY, USA: ACM, 2013. (SA '13), p. 103:1–103:6. ISBN 978-1-4503-2633-9. Disponível em: <<http://doi.acm.org/10.1145/2543651.2543665>>.
- MIXARE. **mixare – Open Source Augmented Reality Engine**. 2014. Disponível em: <<http://www.mixare.org/>>.
- MÖLLER, A. et al. A mobile indoor navigation system interface adapted to vision-based localization. In: **Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia**. New York, NY, USA: ACM, 2012. (MUM '12), p. 4:1–4:10. ISBN 978-1-4503-1815-0. Disponível em: <<http://doi.acm.org/10.1145/2406367.2406372>>.
- MORAN, T. P.; DOURISH, P. Introduction to this special issue on context-aware computing. **Hum.-Comput. Interact.**, 2001. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, v. 16, n. 2, p. 87–95, dez. 2001. ISSN 0737-0024. Disponível em: <http://dx.doi.org/10.1207/S15327051HCI16234_01>.
- MORIMOTO, C. E. **Smartphones – guia prático**. [S.l.]: Editora Sulina, 2009. ISBN 978-85-99593-14-1.
- NARDI, T.; PERES, F. Utilização de realidade aumentada e dispositivos móveis para auxiliar na manutenção de instrumentos de medição de barragens. **1. Ciência da Computação. 2. Congressos. I. Sousa, Thiago C. de (org.) II. Baluz, Rodrigo Augusto RS (org.) III. Título. CDD 001.642**. p. 159.
- PEREY, C.; MIYASHITA, T. Indoor positioning and navigation for mobile ar. In: **Mixed and Augmented Reality - Arts, Media, and Humanities (ISMAR-AMH), 2011 IEEE International Symposium On**. [S.l.: s.n.], 2011. p. 1–1.
- RUBIN, A. **Google's Rubin: Android a revolution**. 2010. Disponível em: <http://news.cnet.com/8301-1023_3-10245994-93.html?tag=mncol>.

- Sá, M. de; CHURCHILL, E. Mobile augmented reality: Exploring design and prototyping techniques. In: **Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services**. New York, NY, USA: ACM, 2012. (MobileHCI '12), p. 221–230. ISBN 978-1-4503-1105-2. Disponível em: <<http://doi.acm.org/10.1145/2371574.2371608>>.
- SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. **Operating systems concepts with Java, 6th Edition**. [S.l.]: Wiley, 2004. ISBN 978-0-471-45249-2.
- SOMMERVILLE, I. et al. **Engenharia de software**. Pearson Prentice Hall, 2008. ISBN 9788588639287. Disponível em: <<http://books.google.com.br/books?id=ifTYOgAACAAJ>>.
- SOUZA, J. O. de; BOTEGA, L. C. Desenvolvimento de navegador de realidade aumentada para exploração regional de zonas de interesse. **REGRAD - Revista Eletrônica de Graduação do UNIVEM**, 2013. v. 6, n. 1, p. 169–174, 2013. ISSN 1984-7866. Disponível em: <<http://revista.univem.edu.br/index.php/REGRAD/article/view/442/334>>.
- STANDARDS, W. **W3C standards**. 2013. Disponível em: <<http://www.w3c.org/standards/>>.
- TANENBAUM, A. **Sistemas operacionais modernos**. Pearson Prentice Hall, 2003. ISBN 9788587918574. Disponível em: <<http://books.google.com.br/books?id=meCAGQAACAAJ>>.
- Fundamentos e Tecnologia de Realidade Virtual e Aumentada.**
- VERTEGAAL, R. Designing Attentive Interfaces. In: **ETRA '02: Proceedings of the symposium on Eye tracking research & applications**. New York, NY, USA: ACM Press, 2002. p. 23–30.
- WOODILL, G. **The Mobile Learning Edge: Tools and Technologies for Developing Your Teams**. McGraw-Hill Education, 2010. ISBN 9780071739849. Disponível em: <<http://books.google.co.nz/books?id=65bKPeCNaTEC>>.