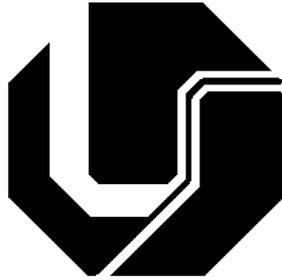


**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**



**UM APLICATIVO COM RECONHECIMENTO DE TOQUES E
GESTOS PARA CONCEPÇÃO DE AMBIENTES VIRTUAIS EM
DISPOSITIVOS MÓVEIS**

Pedro Henrique Cacique Braga

**Outubro
2012**

**UM APLICATIVO COM RECONHECIMENTO DE TOQUES E
GESTOS PARA CONCEPÇÃO DE AMBIENTES VIRTUAIS EM
DISPOSITIVOS MÓVEIS**

Pedro Henrique Cacique Braga

Texto apresentado à Universidade Federal de Uberlândia como parte dos requisitos para obtenção do título de Mestre em Ciências.

Prof. Alexandre Cardoso, Dr.
Orientador

Prof. Edgard A. Lamounier Jr, PhD.
Co-orientador

Prof. Alexandre Cardoso, Dr.
Coordenador do curso de Pós-Graduação

**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**UM APLICATIVO COM RECONHECIMENTO DE TOQUES E GESTOS
PARA CONCEPÇÃO DE AMBIENTES VIRTUAIS EM DISPOSITIVOS
MÓVEIS**

Pedro Henrique Cacique Braga

Texto apresentado à Universidade Federal de Uberlândia, perante a banca de examinadores abaixo, como parte dos requisitos necessários para a obtenção do título de Mestre em Ciências.

Banca Examinadora:

Prof. Alexandre Cardoso, Dr. – Orientador (UFU)

Prof. Edgard A. Lamounier Jr, PhD – Co-orientador (UFU)

Prof. Gilberto Arantes Carrijo, Dr. (UFU)

Prof. Pollyana Notargiacomo Mustaro, Dr. (Universidade Presbiteriana Mackenzie)

DEDICATÓRIA

Este trabalho é dedicado ao autor maior de tudo o que existe: Deus. Que este trabalho seja alegria para Seus olhos, pois dEle é a inspiração, a força e a capacidade para realizá-lo.

“Mas esforçai-vos, e não desfaleçam as vossas mãos; porque a vossa obra tem uma recompensa.” 2 Crônicas 15:7

AGRADECIMENTOS

A Deus, por permitir-me trilhar um caminho de tantas conquistas.

Aos meus pais, José Henrique e Beloní, que são meus maiores exemplos de vida e sem sua dedicação e carinho não poderia chegar onde estou.

Aos meus irmãos, Paulo e Isabela, pelo companheirismo e amizade.

À minha família que sempre esteve presente durante a caminhada.

Aos professores Alexandre Cardoso, Dr. e Edgard Afonso Lamounier Jr., PhD, pelo imenso apoio, não somente para este projeto, mas durante a graduação. Ambos se tornaram grandes exemplos e referência para minha carreira.

Aos colegas de laboratório que serão para sempre amigos e companheiros de carreira.

A todos os amigos que cederam parte do seu precioso tempo para esta conquista.

PUBLICAÇÕES

A seguir, são apresentadas as publicações resultantes deste trabalho.

Braga, P.H.C; Cardoso, A; Lamounier Jr., E.A. **Uma Estratégia de Desenvolvimento de Sistemas para Mobile-Learning usando Realidade Aumentada.** In: Anais do VIII workshop de Realidade Virtual e Aumentada – WRVA'2011, 2011, Uberaba, MG, Brasil.

Braga, P.H.C; Cardoso, A; Lamounier Jr., E.A. **Desenvolvimento de Gestos Personalizados para Criação e Navegação em Ambientes de Realidade Virtual e Aumentada associada a Dispositivos Móveis.** In: Anais do XIV Simpósio de Realidade Virtual e Aumentada - SVR 2012, 2012, Niterói, RJ, Brasil.

RESUMO

BRAGA, Pedro H. C. *UM APLICATIVO COM RECONHECIMENTO DE TOQUES E GESTOS PARA CONCEPÇÃO DE AMBIENTES VIRTUAIS EM DISPOSITIVOS MÓVEIS*, Uberlândia, Faculdade de Engenharia Elétrica – UFU, 2012.

Palavras-chave: M-Learning, Realidade Virtual, Realidade Aumentada

O cenário educacional é constantemente alterado pelo surgimento de novas tecnologias. Diversas pesquisas e ferramentas são desenvolvidas com o propósito de resignificar as formas de aprender e ensinar dentro e fora da escola. As tecnologias da informação e da comunicação causaram uma profunda revolução no modo de vida das pessoas nas últimas décadas. Uma das áreas que não poderia deixar de ser afetada é a da educação e capacitação (Kubota, Demoliner, & Figueiredo, 2011). A Educação ganhou novas perspectivas com a aprendizagem baseada nas tecnologias interativas. O aprendizado torna-se um processo dinâmico e não depende mais das barreiras geográficas. Surgem, com isso, técnicas de ensino, suportadas pela computação móvel, capazes de levar ao usuário novas informações em qualquer lugar e a qualquer momento (*mobile-learning* ou simplesmente *m-learning*). Este trabalho aborda as práticas atuais de *m-learning* e apresenta uma estratégia de desenvolvimento de aplicações educacionais, com uso de dispositivos móveis associados a Ambientes Virtuais e Realidade Aumentada. Para tal desenvolvimento, são adaptadas técnicas de reconhecimento de padrões relacionados com gestos e toques, elaboradas estratégias de associações de tais padrões com elementos de bibliotecas e entre si, visando conceber ambientes virtuais e aumentados como suporte ao ensino. Como prova de conceito, implementou-se e avaliou-se aplicativo que, fundamentado em tais técnicas, permite a concepção de experiências de Física (Mecânica Clássica).

ABSTRACT

BRAGA, Pedro H. C. *Development Strategy for Mobile-Learning Systems using Augmented Reality*, Uberlândia, Faculdade de Engenharia Elétrica – UFU, 2012.

Palavras-chave: M-Learning, Virtual Reality, Augmented Reality

The educational scenario has been constantly changed by the emergence of new technologies. Several researches and tools are developed in order to reframe the ways of learning and teaching inside and outside school. Information and communication technologies have caused a deep revolution in people's way of life in the past decades. One of the ranges that couldn't avoid to be affected is education and enablement (Kubota, DEMOLINER e FIGUEIREDO, 2011). The education has gotten new perspectives with the learning based on interactive technologies. Learning becomes a dynamic process and no more dependent on geographical boundaries. Hence, learning techniques arise, supported by mobile computing, which is able to take new information to the user at any time anywhere (mobile-learning or simply m-learning). This work approaches current usage of m-learning and presents a strategy of development in educational applications, using mobile devices associated with Virtual Environments and Augmented Reality. With that purpose, pattern recognition techniques related to gesture and touch are adapted and strategies of those patterns associations with library elements and between themselves, aiming to conceive virtual and augmented environments as a support for teaching. As a proof of concept, the application was implemented and evaluated, grounded in such techniques, allowing the conception of Physics experiments (Classic Mechanics).

SUMÁRIO

Lista de Figuras	12
Lista de Tabelas e Quadros	14
Lista de Abreviaturas	15
Introdução	1
Fundamentos	6
2.1 Introdução	6
2.2 Fundamentos de Computação Móvel.....	6
2.2.1 Sistemas Operacionais.....	6
2.2.2 Princípios de Design	10
2.2.3 Navegação e Reconhecimento de Gestos.....	13
2.2.4 Estrutura de Aplicativos	16
2.2.5 Compatibilidade e Comunicação	19
2.2.5.1 Compatibilidade	19
2.2.5.2 Comunicação	20
2.3 Realidade Virtual e Aumentada.....	21
Trabalhos Correlatos	25
3.1 Introdução	25
3.2 MIT Magic Paper	25
3.3 In-Place 3D.....	27
3.4 ILoveSketch.....	30
3.5 Resumo Comparativo dos Trabalhos Analisados.....	32
Especificação do Sistema Proposto	34
4.1 Introdução	34
4.2 Requisitos Funcionais	34

4.3	Requisitos Não Funcionais.....	35
4.4	Casos de Uso.....	36
4.4.1	Descrição dos Atores.....	37
4.4.2	Descrição dos Casos de Uso.....	38
4.5	Diagrama de Classes	42
4.6	Pipeline	44
Detalhes da Implementação		45
5.1	Introdução	45
5.2	Estrutura do Aplicativo	45
5.3	Estrutura de Navegação.....	48
5.4	Projeto de Interface	50
5.4.1	Ícones	50
5.4.2	Padrões de Interação.....	51
5.4.2.1	Controle	51
5.4.2.2	Elementos	52
5.4.3	Flexibilidade.....	52
5.5	Criação do Sistema Bidimensional.....	53
5.5.1	Desenho de formas.....	54
5.5.2	Simplificação da forma obtida.....	55
5.5.3	Reconhecimento da forma.....	57
5.5.4	Método alternativo para reconhecimento.....	59
5.5.5	Substituição da Forma Original.....	62
5.6	Sistema de Realidade Virtual	63
5.6.1	Extrusão Linear.....	64
5.6.2	Extrusão por rotação - Sweep	65
5.6.3	Extrusão por caminho definido	65
5.7	Propriedades Físicas do Ambiente Virtual	66
5.8	Sistema de Realidade Aumentada.....	67
5.9	Comunicação entre Dispositivos	67
5.10	Instruções ao Usuário.....	69
5.11	Imagine DT – captação de dados comportamentais	70
Discussão dos Resultados.....		73

6.1	Introdução	73
6.2	Dados coletados pelo ImagineDT	73
Conclusões e Trabalhos Futuros		78
7.1	Introdução	78
7.2	Conclusões.....	78
7.3	Trabalhos Futuros	79
	7.3.1 Aprimoramento do sistema	80
	7.3.2 Reconhecimento de gestos para outras finalidades	80
	7.3.3 M-learning para outras áreas do conhecimento.....	81
7.4	Considerações Finais.....	81
Referências Bibliográficas		82

LISTA DE FIGURAS

Figura 1 – Organização da dissertação.....	5
Figura 2 – Vendas de Smartphones no mundo por Sistema Operacional no primeiro quadrimestre de 2012.....	7
Figura 3 – Dicionário de gestos padrões.....	14
Figura 4 – Dikatação Binária.....	15
Figura 5 – Estrutura padrão de aplicativos para dispositivos móveis.....	17
Figura 6 – Exemplo de tela de transição: Play Store.....	18
Figura 7 - Exemplo de navegação entre telas de edição e detalhes através de gestos: Gmail.....	19
Figura 8 – Exemplos de smartphones com (a) barra de navegação física; (b) barra de navegação virtual.....	20
Figura 9 – Modelo Mental de produto.....	22
Figura 10 – Exemplo de marcador fiducial.....	23
Figura 11 – Magic Paper. (a) Desenho do sistema físico; (b) Animação do modelo.....	26
Figura 12 - InPlace 3D (a) Autoria do sistema mecânico, esboçado no papel; (b) Ambiente de Realidade Aumentada construído com base no modelo desenhado.....	28
Figura 13 – Estrutura do software In-Place 3D.....	29
Figura 14 - ILoveSketch (a) Criação do esboço; (b) Trabalhos finais gerados.....	31
Figura 15 – Diagrama de Casos de Uso.....	37
Figura 16 – Diagrama de classes do sistema.....	43
Figura 17 – Pipeline de utilizações do sistema.....	44
Figura 18 – Estrutura do aplicativo proposto.....	45
Figura 19 – Pipeline estrutural do aplicativo Imagine 3D.....	47
Figura 20 – Estrutura de telas do aplicativo modelo.....	48
Figura 21 – Formas do menu principal: (a) Exposto; (b) Contraído; (c) Aberto.....	49
Figura 22 – Ícones desenvolvidos para o aplicativo Imagine 3D.....	51
Figura 23 – Métrica de um botão.....	52
Figura 24 – Versões do Imagine 3D.....	53
Figura 25 – Passos para simplificação de padrões.....	56
Figura 26 – Simplificação de formas desenhadas.....	56
Figura 27 – Algoritmo \$1 Recognizer, prova de conceito.....	58

Figura 28 – Trecho de código: padrão “S cursiva minúscula”	59
Figura 29 – Codificação dos principais gestos	59
Figura 30 – Dicionário de sinais	60
Figura 31 – Processo de codificação de gestos.	61
Figura 32 – Telas de substituição de formas.	62
Figura 33 – Edição de formas bidimensionais	63
Figura 34 – Métodos de extrusão: (a) linear; (b) rotação; (c) e (d) caminho.....	64
Figura 35 – Tela de visualização do ambiente virtual.....	66
Figura 36 – (a) Comunicação entre dispositivos e (b) compartilhamento de modelos.	68
Figura 37 – Vídeo com instruções de uso do Imagine 3D	70
Figura 38 – Interface do aplicativo de coleta de dados: Imagine DT	71
Figura 39 – Diagrama de Classes do Banco de Dados.....	72
Figura 40 – Visualização dos dados obtidos: Padrões por Usuário	74
Figura 41 – Gráficos de tempo e pontos por usuário	76
Figura 42 – Gráfico de porcentagem de reconhecimento por padrão	76

LISTA DE TABELAS E QUADROS

Tabela 1 – Vendas mundial de smartphones no 1º quadrimestre de 2012, por sistema operacional.....	7
Tabela 2 – Quadro comparativo entre os trabalhos correlatos.....	33
Tabela 3 – Dados dos usuários do Imagine DT	75
Quadro 1 - Atores do sistema.....	38
Quadro 2 - Descrição do caso de uso “Registrar Usuário”	38
Quadro 3 - Descrição do Caso de Uso “Editar Perfil”	39
Quadro 4 - Descrição do Caso de Uso “Criar Esboço 2D”	40
Quadro 5 - Descrição do Caso de Uso “Compartilhar Esboço”	40
Quadro 6 - Descrição do Caso de Uso “Gerar Ambiente Virtual”	41
Quadro 7 - Descrição do Caso de Uso “Assignar Propriedades Físicas ao Modelo”	41
Quadro 8 - Descrição do Caso de Uso “Visualizar Modelo em RA”	42
Quadro 9 - Descrição do Caso de Uso “Comunicar-se com o Grupo”	42

LISTA DE ABREVIATURAS

ANATEL	Agência Nacional de Telecomunicações
API	<i>Application Programming Interface</i>
APP	Aplicativo
AS3	<i>Actionscript 3.0</i>
AVA	Ambiente Virtual de Aprendizagem
BD	Banco de Dados
CGI	Comitê Gestor de Internet
EAD	Educação a Distância
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
MEC	Ministério da Educação
M-Learning	<i>Mobile Learning</i>
PDI	Processamento Digital de Imagens
RA	Realidade Aumentada
RV	Realidade Virtual

"A imaginação é mais importante que o conhecimento. O conhecimento é limitado. A imaginação envolve o mundo."

- Albert Einstein

CAPÍTULO 1

INTRODUÇÃO

O cenário educacional é constantemente alterado pelo surgimento de novas tecnologias. Diversas pesquisas e ferramentas são desenvolvidas com o propósito de resignificar as formas de aprender e ensinar dentro e fora da escola. As tecnologias da informação e da comunicação causaram uma profunda revolução no modo de vida das pessoas nas últimas décadas. Uma das áreas que não poderia deixar de ser afetada é a da educação e capacitação (Kubota, Demoliner, & Figueiredo, 2011).

“A informação representa o principal ingrediente de nossa organização social, e os fluxos de mensagens e imagens entre as redes constituem o encadeamento básico de nossa estrutura social” (Castells, 1999).

A aprendizagem é um processo tão antigo quanto à própria existência humana e se desenvolve constantemente mediante as contribuições de pesquisadores e educadores. O processo de aprendizagem sofre alterações ao longo do tempo e é diferenciado conforme as posições geográfica e social de um grupo de indivíduos. A fim de obter melhor aproveitamento de tal processo faz-se necessário adequar constantemente as práticas de ensino ao contexto em que estas estão inseridas.

O processo cognitivo do ser humano pode beneficiar-se com a associação de todos os seus sentidos na construção de um conceito, o que inspira a constante busca de métodos de ensino que proporcionem maiores possibilidades interativas e desafiadoras para aqueles que os utilizam. Cognição é uma complexa coleção de funções mentais que incluem atenção, percepção, compreensão, aprendizagem, memória, resolução de problemas e raciocínio, entre outras, que permitem que o homem compreenda e relacione-se com o mundo e seus elementos (Jesus, Fernandes, Dazzi, & Moreira, 2008).

Nesse contexto, destaca-se a busca por metodologias de ensino mais adequadas ao contexto social e aos avanços que a sociedade apresenta. Diante desse cenário é possível dizer que Educação e Tecnologia se conectam, interagem

no processo educacional evidenciando a necessidade de desenvolvimento de ferramentas compatíveis as novas exigências.

Com os novos modelos de aprendizagem, o conhecimento deixou de ser percebido como absoluto para se relativizar em cada momento, ante os novos avanços científicos e tecnológicos (Feldmann, 2005).

A Educação a Distância já é uma modalidade de ensino presente no Brasil e em constante crescimento. Os cursos de educação a distância aumentaram 30,4% enquanto os presenciais 12,5% (MEC, 2010).

Com o intuito de atender a tais exigências do progresso, foram criados diversos métodos e ferramentas de ensino que realizam a conexão entre Tecnologia e Educação, como jogos educativos, ambientes virtuais de aprendizagem, sistemas de apoio didático e consulta eletrônica, por exemplo. Ambientes Virtuais de Aprendizagem (AVAs) são sistemas computacionais utilizados para o Ensino a Distância. Estes ambientes fazem parte do conjunto de técnicas do modelo de ensino não presencial, conhecido como e-learning.

Para suportar os recursos de EAD e difundí-los ainda mais, o acesso do aluno à tecnologia deve acompanhar esse desenvolvimento. Um computador com acesso à Internet é requisito mínimo para participar de um curso a distância.

Segundo o Comitê Gestor de Internet (CGI), entre 2009 e 2010 a proporção de brasileiros no Brasil com computador passou de 32% para 35%, representando um crescimento no acesso à Internet de 9%. Em 2010 o acesso à Internet nos domicílios urbanos cresceu 15% em relação ao ano anterior. Em 2009 a porcentagem de usuários de cursos online sobre o total de internautas foi de 11% (Campos, 2006).

Quando este modelo é utilizado em dispositivos móveis, passa a ser conhecido como *mobile-learning*, ou simplesmente *m-learning*, uma prática que combina recursos de telecomunicação móvel com as premissas de *e-learning*, nas quais tecnologias baseadas em computação são usadas no processo educacional, provendo treinamento individualizado, disponível a qualquer momento e em qualquer lugar (Tucker & Winchester III, 2009).

Paralelo ao cenário educacional, o cenário tecnológico está cada vez mais voltado para os dispositivos móveis. No final de 2011, o Brasil contava com 242,2 milhões de acessos do Serviço de Telefonia Móvel Pessoal, registrando um crescimento de 19,4% em relação ao ano anterior (ANATEL, 2011).

A transferência das práticas de e-learning para dispositivos móveis é inerente ao crescimento de seu uso e apresenta-se de forma natural. Contudo, assim como foi necessário ser estudada com cautela a transição de ambientes presenciais de aprendizagem para ambientes virtuais, é necessário criar novos paradigmas de apresentação do conteúdo em novos dispositivos.

O desenvolvimento de aplicativos para m-learning requer estudos profundos sobre usabilidade, apresentação do conteúdo e design de interfaces. É preciso recriar paradigmas de design, pois ainda são usados antigos paradigmas para desktop na criação dos aplicativos, não aproveitando ao máximo os recursos dos dispositivos (Hooper & Berkman, 2012).

Entre os inúmeros recursos dos smartphones, tablets e outros dispositivos móveis, a navegação em ambientes tridimensionais tornou-se alvo de pesquisas. A busca pela fluidez da navegação e da melhor maneira de serem utilizados os recursos de processamento dos dispositivos coloca em evidência a necessidade de melhores técnicas de interação. O constante desenvolvimento de aplicativos com ambientes 3D, principalmente voltados para jogos, requer um estudo sobre a usabilidade de tais sistemas. O foco das pesquisas não está somente na navegação, mas na criação interativa e colaborativa dos ambientes virtuais.

O objetivo central deste trabalho é propor um aplicativo de m-learning, associado a interfaces naturais, que suporta o reconhecimento de gestos e toques e a concepção de ambientes virtuais e aumentados.

Considerando dispositivos de computação móvel, são objetivos específicos deste trabalho:

1. Conceber uma interface que suporte reconhecimento de gestos e toques, inferindo formas e as associando com uma biblioteca de modelos previamente definida;
2. Elaborar e aperfeiçoar algoritmos rápidos de captura e reconhecimento de gestos e toques para tais dispositivos;
3. Propor práticas para criação de ambientes virtuais a partir de esboços bidimensionais;
4. Elaborar um aplicativo, como prova de conceito, que suporta o desenvolvimento de experimentos de Física (Mecânica Clássica);

Para a concretização deste objetivo, foram traçadas as seguintes metas:

1. Estudo consiso da criação de aplicativos mobile, focando principalmente nas formas de interação atuais, design funcional de interfaces e seus diferentes paradigmas;
2. Desenvolvimento de algoritmos eficazes para reconhecimento e interpretação de gestos;
3. Elaboração de métodos para transição de esboços bidimensionais para elementos tridimensionais que respeitem suas características geométricas;
4. Desenvolvimento de um sistema de simulação de Física para interação entre os elementos 3D, o ambiente virtual e a interação do usuário;

Esta dissertação é estruturada em sete capítulos distribuídos conforme Figura 1.

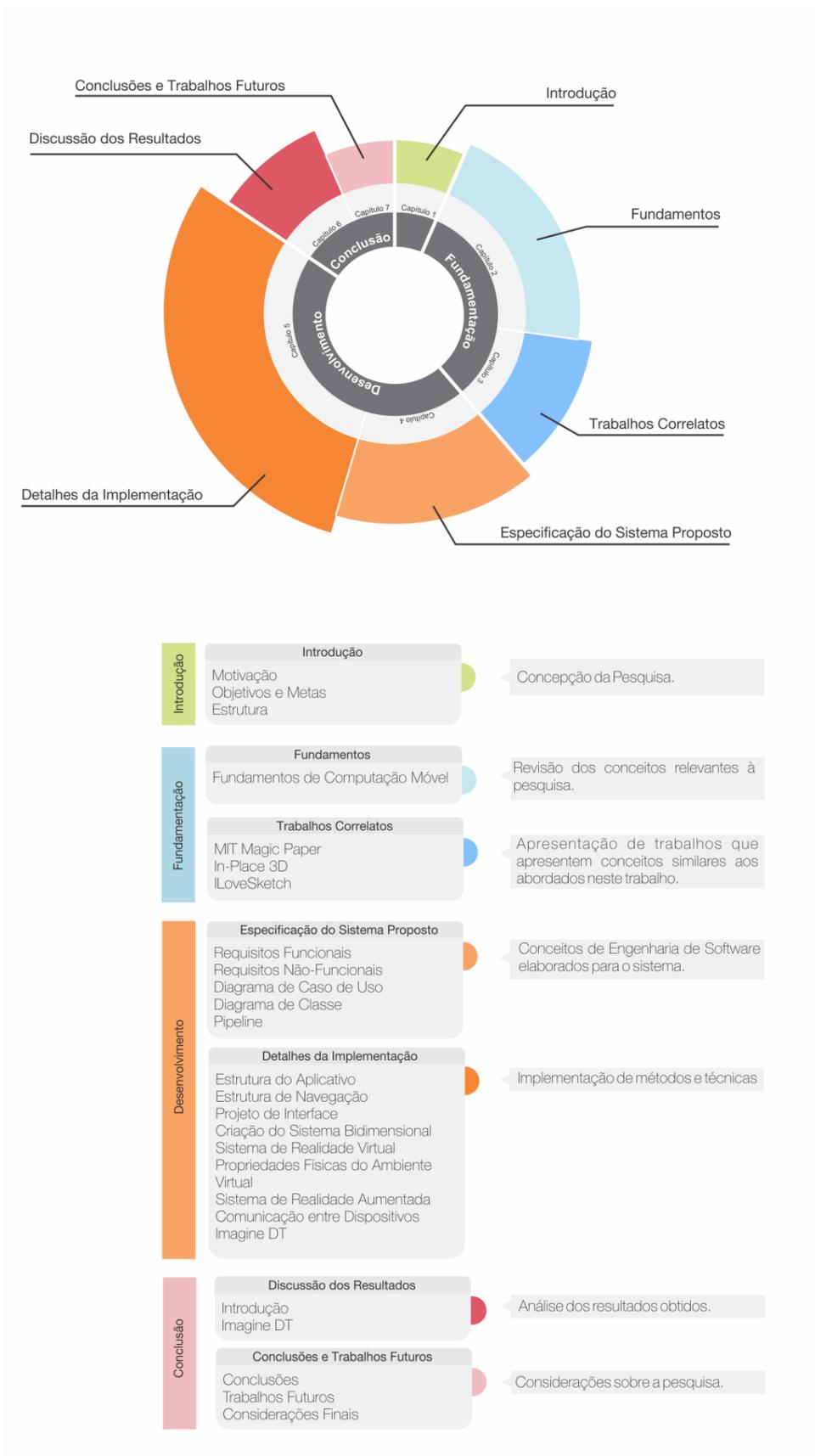


Figura 1 – Organização da dissertação

CAPÍTULO 2

FUNDAMENTOS

2.1 Introdução

Neste capítulo serão abordados tópicos que fundamentam a pesquisa sobre computação móvel, *m-learning* e Realidades Virtual e Aumentada.

2.2 Fundamentos de Computação Móvel

2.2.1 Sistemas Operacionais

Os computadores tem um dispositivo de software denominado sistema operacional, 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, 2008).

Dispositivos móveis não são diferentes neste aspecto e necessitam um sistema operacional para gerenciar seus recursos de hardware e proporcionar ao usuário a melhor experiência possível em usabilidade.

As grandes empresas provedoras de SO's para computadores pessoais apostam também em sistemas para dispositivos móveis, principalmente os *smartphones* e *tablets*. Entre os principais, encontram-se os sistemas *Android*, *iOS* e *Symbian*, mas outros vem sendo desenvolvidos e buscam sua parcela de mercado. O surgimento de novos aparelhos é constante e a cada novo dispositivo desenvolvido, os SO's móveis ganham sua força.

Segundo pesquisa realizada pela Gartner Inc., no primeiro quadrimestre de 2012, das vendas de *smartphones* pelo mundo, cerca de 56% operam com o sistema *Android*, enquanto que 22% conta com o sistema da *Apple*, *iOS*. O sistema *Symbian OS* conta com aproximadamente 8% das vendas e a parcela restante se divide entre os demais sistemas operacionais (Gartner, 2012).

A Figura 2 mostra o número de *smartphones* vendidos no primeiro quadrimestre de 2012, agrupados pelo sistema operacional utilizado. É perceptível que uma grande parcela dos mesmos é dominada pelo sistema *Android*, seguido pelos sistemas *iOS* e *Symbian*. Vale ressaltar que por ser um sistema de código aberto, o *Android* apresenta várias versões e não se limita a apenas um fabricante de *smartphones*, enquanto que o *iOS* é de propriedade da *Apple*, portanto apenas seus aparelhos contam com o sistema nativo.

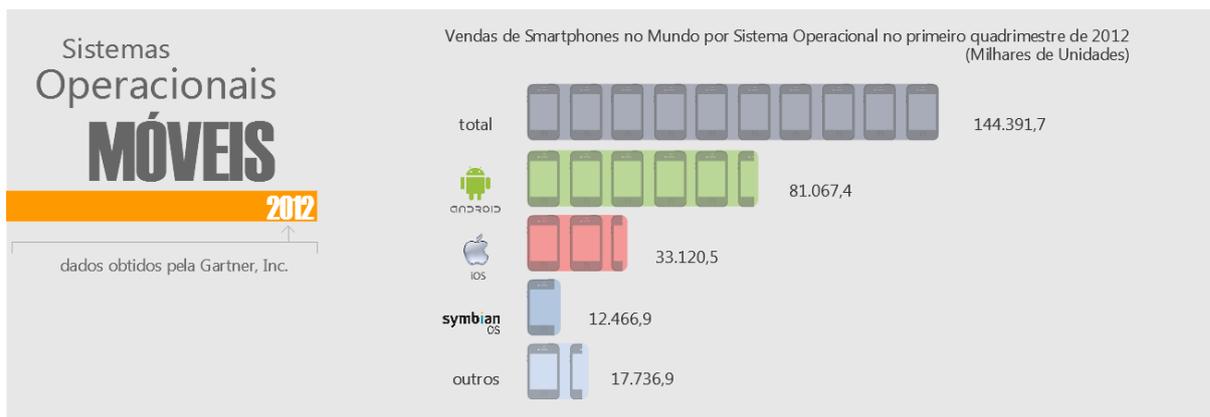


Figura 2 – Vendas de Smartphones no mundo por Sistema Operacional no primeiro quadrimestre de 2012

Fonte: Adaptado de (Gartner, 2012)

A Tabela 1 apresenta detalhadamente as vendas mundiais de *smartphones* por sistema operacionais. Os dados foram obtidos também pela Gartner (Gartner, 2012) e mostram os demais SO's que competem pelo mercado de sistemas operacionais móveis.

Tabela 1 – Vendas mundial de smartphones no 1º quadrimestre de 2012, por sistema operacional.

Sistema Operacional	Unidades vendidas em 1Q12
Android	81.067,4
iOS	33.120,5
Symbian	12.466,9
Research in Motion	9.939,3
Bada	3.842,2
Microsoft	2.712,5
Others	1.242,9
Total	144.391,7

A seguir são apresentadas as principais características dos sistemas mais populares:

- *Android*

Android é a plataforma móvel mais popular do mundo. Com o SO é possível usar os aplicativos do Google usados no computador pessoal, além de mais de 600.000 *apps* e jogos disponíveis no *Google Play*. São disponíveis também milhões de músicas e livros e centenas de filmes. Os dispositivos *Android* são também considerados inteligentes, mantendo o foco no usuário, colocando-o no controle da experiência em mobilidade (Google Inc., 2012).

O *Android* é um sistema de código aberto, o que permite que muitos desenvolvedores aproveitem ao máximo de suas funcionalidades ao criar novos *apps*¹. A loja virtual de aplicativos é conhecida como *Google Play* e mantém disponíveis não somente os aplicativos oficiais criados pela empresa, mas *apps* criados por qualquer desenvolvedor cadastrado. Observando as políticas de criação é possível desenvolver aplicativos gratuitos ou mesmo cobrar por um *app* mais completo ou por funções adicionais do *app* gratuito.

A cada dia mais de um milhão de novos aparelhos *Android* são ativados no Mundo. Usuários do *Android* fazem *download* de mais de 1.5 bilhões de aplicativos e jogos na *Play Store* a cada mês (Google Inc., 2012).

Além de fornecer o código de maneira aberta aos desenvolvedores, o Google disponibiliza também APIs para desenvolvimento de novos recursos. Também existem ferramentas de criação de aplicativos simples para não programadores, tornando o SO ainda mais acessível, dando ao usuário comum a possibilidade de criar seus próprios *apps*.

Baseado em Linux, o sistema conta com suporte às conexões mais utilizadas pelos dispositivos, como 3G, WI-FI e EDGE, por exemplo. Fabricantes como Acer, HP, HTC, Motorola e Sony Ericson adotam o SO em seus aparelhos.

- *iOS*

O Sistema desenvolvido especialmente para os aparelhos da Apple é o segundo mais utilizado segundo o número de vendas de dispositivos Móveis. Com uma

¹ Os softwares para SO's móveis são conhecidos como aplicativos, ou simplesmente *apps*.

interface de fácil utilização, incríveis características, bastante estável, *iOS* é o fundamento do iPhone (Apple, About iOS, 2012).

Desenvolvido por um dos grandes nomes da história da computação – Steve Jobs – e sua equipe, o SO está em sua versão 5, e já conta com a versão 6 em desenvolvimento. Com foco no design, os aparelhos da Apple alcançam uma parcela cada vez maior dos usuários de *smartphones*.

Jobs achava que um elemento central da simplicidade do design era fazer produtos intuitivamente fáceis de usar. “O principal no nosso design é que temos de fazer as coisas intuitivamente óbvias” (Isaacson, 2011).

Uma grande marca do sistema, bem como dos dispositivos Apple é a usabilidade. De acordo com os princípios de design criados por Jobs, 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 inúmeros *apps* gratuitos e/ou pagos. O desenvolvimento de aplicativos é mais limitado, devido aos direitos da empresa sobre o sistema *iOS*. Para tal, faz-se necessário o uso de softwares exclusivos e, oficialmente, o uso do sistema operacional para desktop *MAC OS*. Uma linguagem específica, conhecida como *Objective-C* foi criada, baseada na linguagem *C*, para a programação de aplicativos do *iOS*.

- Symbian OS

O Symbian é fruto da parceria entre Ericsson, Nokia, Motorola e PSION, e foi criado para ser uma plataforma aberta para aplicativos. Em outras palavras, uma plataforma na qual qualquer um pode criar e compartilhar softwares instaláveis, aplicações para usuários finais, utilitários ou extensões para serviços e outras facilidades (Morris, 2007).

Apesar do constante desenvolvimento de novas versões do sistema, ele vem perdendo mercado para os gigantes *Android* e *iOS*. Entretanto, ainda há fabricantes que optam por ele, como um sistema alternativo de seus aparelhos. A Nokia, por exemplo, mantém o SO como padrão, mas vem explorando outros sistemas, como o *Android*, a fim de manter-se atualizada no mercado.

Outros sistemas estão em ascensão, como o *Windows Phone*, por exemplo, que disputam por uma parcela maior no mercado. Apesar de conquistar cada vez mais

usuários, não se mostra como grande ameaça aos grandes competidores citados acima.

2.2.2 Princípios de Design

Todo software criado precisa de uma interface entre o usuário e os comandos de máquina. Tal interface deve ser bem desenhada a fim de obter melhor aproveitamento do software e do uso do hardware disponível. Pensando em softwares desenvolvidos para computadores pessoais, existem diversos padrões de design interface de sucesso. Pode-se citar, por exemplo, o padrão *Windows* (Microsoft, 2012). Esse padrão é baseado no uso de janelas, que são sub-interfaces do mesmo sistema direcionadas a uma ação específica. Por se tratar de um dos padrões mais antigos, é o mais utilizado em computadores pessoais e por isso o usuário comum assimila rapidamente suas funções.

Este padrão tem diferentes versões que se baseiam nos sistemas operacionais existentes, mas o conceito fundamental é o mesmo: o uso de janelas.

Em se tratando de dispositivos móveis, percebe-se que esse padrão não é tão eficaz. Entretanto, uma coisa que ainda não aconteceu é uma padronização global do design de interfaces móveis (Hoover & Berkman, 2012).

Um ponto em comum entre os novos paradigmas é o uso de telas ao invés de janelas. As telas se comportam de maneira dinâmica e a transição entre elas pode ser explorada de forma fluida. A principal diferença entre janelas e telas é a forma com que são apresentadas e os métodos de transição entre elas. Este conceito será abordado novamente no desenvolvimento do aplicativo para prova de conceito.

Pensando nas tecnologias de interação desenvolvidas para dispositivos móveis, devem ser pensados novos paradigmas de design.

O *Google* tem buscado, através das comunidades de desenvolvedores para *Android*, padronizar a criação de aplicativos. Os trabalhos de projeto de interfaces para *Android* são focados em três elementos principais: beleza, simplicidade e usabilidade (Google Inc., 2012).

Um bom projeto de interface móvel começa com a clara definição da funcionalidade do *app*². Usuários do *iOS* são acostumados à aparência e ao bom

² Aplicativos, ou simplesmente *apps* são softwares criados para dispositivos móveis

comportamento dos *apps*, assim eles tendem a esperar experiências similares aos próximos aplicativos instalados (Apple, iOS Developer Library, 2012).

- Beleza

O usuário busca ser surpreendido por uma superfície bonita, com animações cuidadosamente inseridas e sons adicionados a momentos específicos da navegação.

Objetos reais são mais bem aceitos que botões comuns. Ao permitir o usuário tocar diretamente e manipular um objeto, o esforço cognitivo necessário é menor para completar uma tarefa e o envolvimento emocional é maior (Google Inc., 2012).

É desejável que um *app* seja personalizável, de maneira que o usuário possa alterar elementos de design, como cores e imagens de fundo. A personalização, associada ao aprendizado de máquina garante a melhor aceitação do aplicativo e a prolongação de seu uso (Google Inc., 2012).

- Simplicidade

As pessoas tendem a pular a leitura de frases compridas. Desta forma, é preferencial o uso de pequenas sentenças e/ou imagens para explicar um procedimento.

Imagens capturam a atenção do usuário e traduzem uma idéia de maneira mais eficiente que palavras (Alun-Jones, 2010).

Usando técnicas de inteligência artificial, é possível diminuir o número de ações tomadas pelo usuário. Com o uso contínuo do aplicativo é possível tomar pequenas decisões sem que o usuário tenha que confirmá-las a cada instante. Entretanto é de boa prática deixar sempre disponíveis atalhos para desfazer uma ação automática, garantindo ao usuário o controle da navegação.

As pesquisas de desenvolvimento de interface realizadas pelo Google mostraram que o usuário prefere visualizar em cada tela apenas o necessário, dispensando informações adicionais (Google Inc., 2012).

Também é importante manter um mapa do aplicativo. Um *app* bem estruturado mantém um padrão de telas comum, geralmente com duas ou três ramificações de tela no máximo.

Manter uma memória *cache* da navegação do usuário é importante para o caso de um fechamento inesperado. Deve-se também salvar configurações a cada

uso do *app* como toques pessoais e até mesmo configurações do aplicativo para um usuário em diferentes dispositivos.

Por fim, segundo os manuais de criação de aplicativos elaborados pelos fabricantes, as interrupções do *app* são inadmissíveis durante a sua utilização, ao menos que sejam de suma importância para seu uso. Deve-se tomar o cuidado de habilitar ou desabilitar as chamadas de outros aplicativos durante o uso.

- Usabilidade

É de boa prática fornecer ao usuário ao menos em seu primeiro acesso, dicas de como utilizá-lo e quais os atalhos existentes para cada tarefa. Uma apresentação dos padrões gráficos adotados deve ser feita no primeiro acesso e permanecer ativa caso o usuário deseje.

A correção de erros ocorridos deve ser feita sempre que possível de maneira automática e imperceptível ao usuário. Tendo em vista que os aparelhos são considerados inteligentes, espera-se um comportamento semelhante ao procedimento humano. Sendo assim, esperam-se mensagens de erro escritas nos padrões coloquiais do idioma adotado.

Atividades complexas podem ser preferencialmente divididas em pequenos passos para serem melhor assimiladas. É importante manter um *feedback* em cada etapa, ou seja, informar o sucesso ou falha das mesmas (Google Inc., 2012).

Nem todas as ações são de igual importância. Aquelas de maior relevância devem ser indicadas de forma rápida e com padrões gráficos de simples assimilação.

Outros padrões adotados para o desenvolvimento, como tamanhos ótimos, sistemas de navegação e estrutura de telas, por exemplo, devem ser observados nos guias de cada sistema operacional.

2.2.3 Navegação e Reconhecimento de Gestos

O mapeamento de ações realizadas no mundo real para ambientes virtuais é um processo complexo. Navegação e seleção de objetos são dois exemplos clássicos. O desafio de projetos de Interfaces Naturais (NUI³) para ambientes 3D é um trabalho difícil devido ao fato de haver modelos extremamente acurados, baseados em modelos físicos (Alun-Jones, 2010).

Os seres humanos usam, naturalmente, de gestos para se comunicar. Tem sido demonstrado que crianças podem facilmente aprender a se comunicar com o gesto antes de aprender a falar. Um gesto é uma comunicação não verbal realizada com uma parte do corpo humano (Bhuiyan & Picking, 2009).

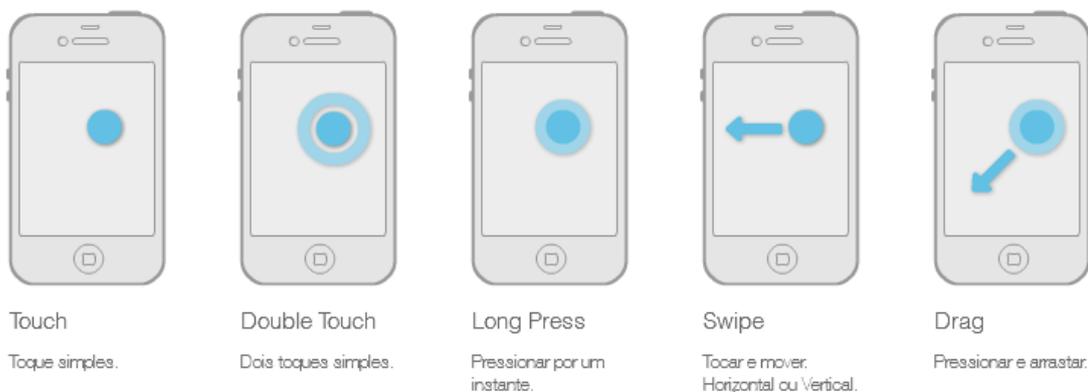
Telas de toque crescem rapidamente em popularidade como um método de entrada para *smartphones* e outros dispositivos móveis, que apresentam poucos botões físicos, ou mesmo nenhum deles. Em vez disso existe o paradigma dominante de invocação de comandos por telas de toque, o que é eficaz quando o usuário está focado diretamente no aparelho, sem distrações (Bragdon, Nelson, Li, & Hinckley, Experimental Analysis of Touch-Screen gesture Designs in Mobile Environments, 2011).

Um gesto é uma combinação de toque e/ou movimento associados a um comando específico de controle ou navegação do aplicativo. Neste trabalho são abordados os gestos de toque e multitoque, isto é, gestos criados a partir do pressionar e arrastar de um ou mais dedos sobre a tela sensível.

Este tipo de interação é comum aos *apps* disponíveis para a maioria dos *smartphones* e suas funções são programáveis de acordo com o intuito de cada aplicativo. Cada SO conta com seus gestos padrões, mas um dicionário de gestos comuns à maioria dos sistemas pode ser visto na Figura 3, que apresenta o conjunto de gestos aceitos pelo *Android* (Google Inc., 2012).

³ NUI – Natural User Interfaces

Gestos com um dedo



Gestos com dois dedos

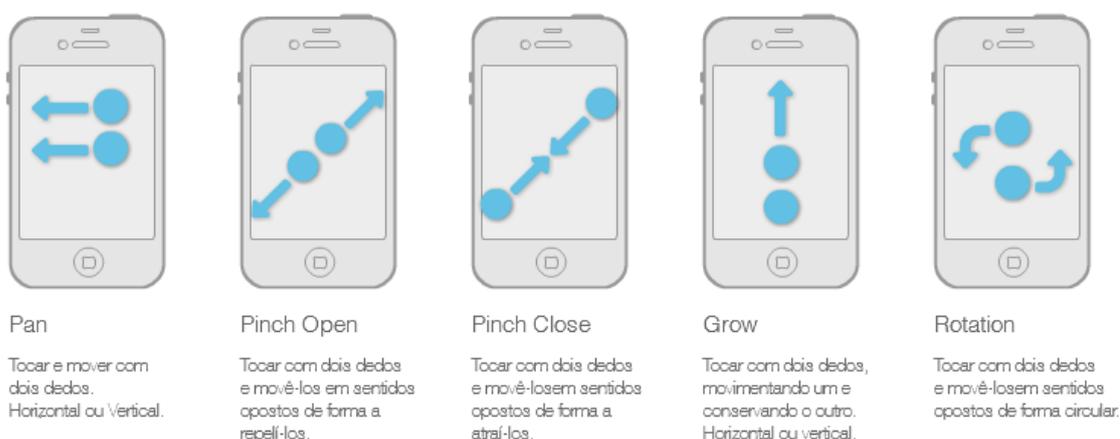


Figura 3 – Dicionário de gestos padrões

Fonte: Adaptado de (Google Inc., 2012)

A popularização dessa tecnologia requer um estudo significativo em busca de melhorias para seu funcionamento ser cada vez mais natural ao usuário. Sendo assim, muitas pesquisas surgem para aprimoramento do reconhecimento dos gestos e padrões (Wobbrock, Wilson, & Li, 2007).

Estudos nas áreas de Reconhecimento de Padrões e Processamento Digital de Imagens têm se intensificado nos últimos anos. Os métodos utilizados para detecção e reconhecimento de padrões podem ser classificados de diferentes maneiras, como métodos de busca por morfologias matemáticas, extração de características, análises de modelos de cores, etc (Shi-Hong, Hong, Chin, Ming, & Yao, 1998).

O reconhecimento de padrões é uma tarefa de elevado nível de complexidade e requer ferramentas computacionais adequadas. Métodos de Inteligência Artificial como Algoritmos Genéticos ou Redes Neurais Artificiais, por exemplo, garantem bons resultados na aproximação de resultados e otimização de sistemas.

Um método bastante eficiente no reconhecimento de padrões, sobretudo em imagens é baseado na morfologia matemática. O estudo morfológico concentra-se na estrutura geométrica das imagens e pode ser aplicado em diversas áreas de processamento de imagens, como realce, filtragem, segmentação, esqueletização e outras afins.

A palavra Morfologia significa forma e estrutura de um objeto ou os arranjos e inter-relacionamentos entre as partes de um objeto. Quando se pensada em métodos digitais, trata-se de um caminho para descrever ou analisar a forma de um objeto digital.

A morfologia trata imagens com base nas teorias de conjuntos matemáticos e aplica sobre elas operações como Intersecção, união, abertura, fechamento, entre outros. O reconhecimento de padrões usando estes métodos é um conjunto de operações sequenciais que extraem características únicas de uma imagem e as comparam com um banco de dados existente.

Uma maneira de se pensar o reconhecimento de gestos é analisa-los como uma imagem que descreve um percurso. A aplicação de filtros morfológicos pode delimitar os principais pontos de interesse do padrão, como curvas acentuadas e arestas soltas, por exemplo. A Figura 4 exemplifica uma operação de Dilatação Binária. O filtro, quando aplicado, pode expandir o limite da imagem, a fim de separar ruídos da imagem.

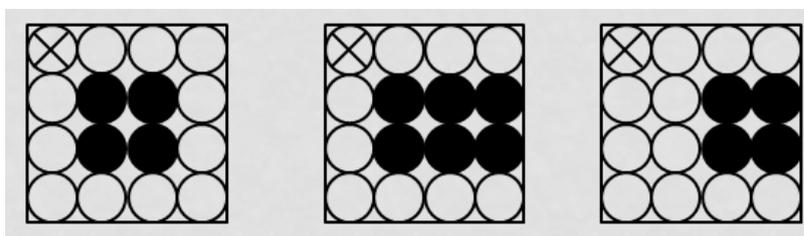


Figura 4 – Dilatação Binária

Fonte: Adaptado de (Gonzalez & Woods, 2000)

Cada imagem submetida a um operador morfológico resulta em uma nova imagem com características únicas. Quando submetidas sequencialmente a outros operadores, apresenta pontos que as distinguem de outras imagens semelhantes.

No domínio dos padrões gestuais, podem ser usados operadores morfológicos que examinam um conjunto de pontos no espaço, tratando-os e selecionando atributos como quantidade e localização de curvas, padrões retos repetidos em sequencia, orientação do gesto, etc.

Outra abordagem possível seria a aplicação de filtros no domínio da frequência para a detecção de bordas na imagem gerada pela introdução do gesto e em seguida, a análise de cada trecho das bordas, produzindo uma sequencia de códigos que indicam movimento.

Diferentes algoritmos para reconhecimento dos padrões de entrada por toque e gesto têm sido desenvolvidos a fim de obter melhores resultados em termos de tempo de resposta e fidelidade do reconhecimento.

Bragdon (2011) apresenta um estudo comparativo entre métodos de interação por toque utilizando diferentes dispositivos. São feitas comparações em termos de velocidade de reconhecimento de padrões bem como análises de usabilidade de gestos em *smartphones*. Lü e Li (2011) descrevem métodos de criação e reconhecimento de gestos usados para navegação em *web browsers* para dispositivos móveis. Por fim, Wobbrock (2007) apresenta o *\$1 Dollar Recognizer*, um método de reconhecimento sem uso de bibliotecas e *toolkits*⁴.

Neste trabalho serão analisados alguns dos principais algoritmos e a combinação dos mesmos na tentativa de criação de gestos personalizados.

2.2.4 Estrutura de Aplicativos

A diversidade de temas e estilos de aplicativos é proporcional ao número de aplicativos existentes nos mercados online. Entretanto, uma estrutura padrão deve ser mantida, para que o processo de aprendizagem de um novo *app* seja simples e intuitivo.

Ao optar por um novo paradigma de design que não o Windows, faz-se necessária a criação de um novo conjunto de regras e modelos de criação. O desenvolvimento de *apps* tem seguido um esquema de criação de telas em substituição das janelas do modelo Windows e a rede formada pelas telas apresenta certo padrão seguido pelos desenvolvedores. A Figura 5 apresenta o modelo de

⁴ *Toolkit* é um conjunto de *widgets*, elementos básicos de uma *GUI*.

estrutura de aplicativos proposto pelo Google em seu guia de design de interfaces (Google Inc., 2012).

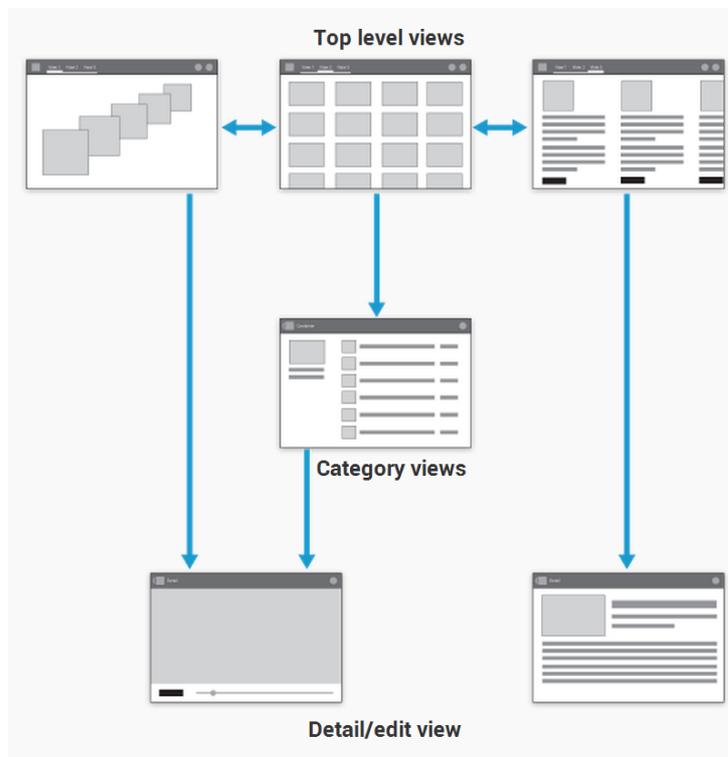


Figura 5 – Estrutura padrão de aplicativos para dispositivos móveis

Fonte: (Google Inc., 2012)

Um *app* típico do *Android* consiste em **telas de nível superior** (*top level views*) e **telas detalhadas de edição** (*Detail/edit view*). Caso a hierarquia da navegação seja profunda e complexa, são necessárias as **telas de transição** (*Category views*) para a conexão das principais.

Top level views são telas que tipicamente apresentam as funcionalidades que o aplicativo suporta. Podem apresentar diferentes visões do mesmo conjunto de dados. *Category views* permitem a navegação entre as telas de nível superior e as detalhadas. Por fim, *detail/edit views* são telas onde o usuário pode criar ou customizar os dados do aplicativo, bem como as telas de configuração do mesmo.

Torna-se importante ressaltar a hierarquia presente entre as telas. É possível navegar de forma horizontal, isto é, em um mesmo nível de telas e verticalmente, passando de um nível para outro com ou sem auxílio das telas de transição.

Para as telas de nível superior, recomenda-se o uso de elementos simples, menor quantidade de texto e apresentação apenas de informações essenciais, mantendo o design do aplicativo e a informação principal como focos da tela.

São bastante utilizadas barras de navegação, que apresentam ícones de fácil associação do desenho à sua função. Tais barras facilitam a transição entre telas e reforça a identidade visual do aplicativo.

Telas de transição geralmente apresentam pequenas tabelas com introduções aos conteúdos finais e ao mesmo tempo métodos de seleção para a navegação. A Figura 6 apresenta um caso de estruturação de tabelas em uma tela de categoria presente na *Play Store*. Observa-se a qualidade dos gráficos apresentados que mantém a identidade visual do Google Inc. A barra de navegação neste caso é utilizada para navegar horizontalmente entre as telas de transição. Foram usadas palavras curtas em vez de ícones para não poluir graficamente a interface.

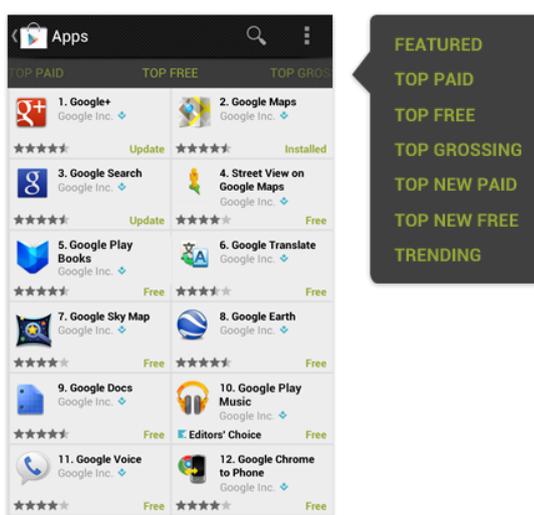


Figura 6 – Exemplo de tela de transição: Play Store

Fonte: (Google Inc., 2012)

As telas de edição geralmente possuem formatação mais simples, fazendo uso de conteúdo secundário fora de foco para concentrar a atenção do usuário no conteúdo relevante. Google (2012) sugere e enfatiza o uso de gestos para navegação mais simples e prática entre telas de edição.

A Figura 7 mostra um exemplo eficiente de navegação entre telas de edição do Gmail. O uso de gestos como o SWIPE garante que o usuário possa transitar de uma tela para outra de forma intuitiva e clara.

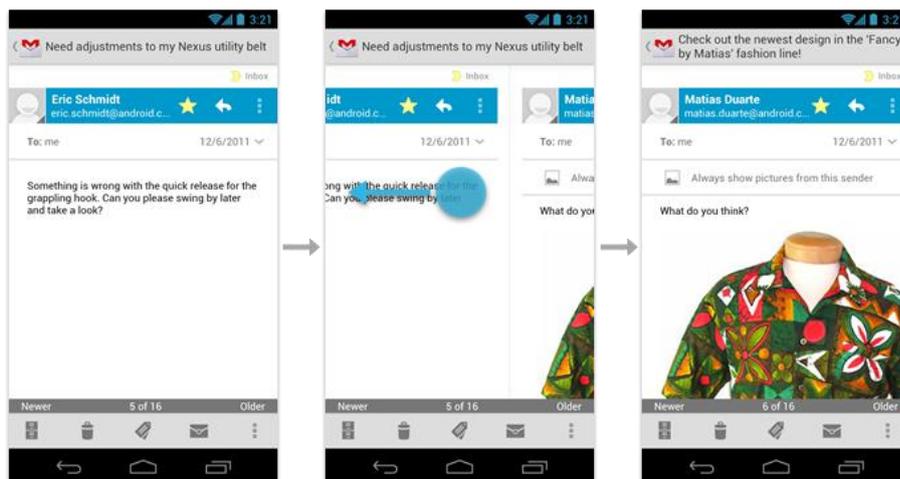


Figura 7 - Exemplo de navegação entre telas de edição e detalhes através de gestos: Gmail.

Fonte: (Google Inc., 2012)

2.2.5 Compatibilidade e Comunicação

2.2.5.1 Compatibilidade

Com a diversidade de aparelhos e versões dos sistemas operacionais disponíveis, é preciso criar os aplicativos de forma a alcançar o maior número de usuários possível. Para tal, é preciso ter sempre atenção a detalhes de implementação que podem ser bem diferentes em modelos de *smartphones* diferentes.

Um exemplo de compatibilidade de hardware que se sobressai é a quantidade de botões físicos que os aparelhos possuem. Até então, a evolução dos dispositivos caminha para a inexistência de botões físicos, deixando apenas gestos e toque para navegação.

Grande parte dos aparelhos com o sistema *iOS* possuem apenas botões laterais para controle de áudio, alternar o estado do telefone entre ligado e desligado e um botão na parte frontal para controle do menu.

Para aparelhos controlados por *Android*, os botões variam entre os mesmos laterais citados e na parte frontal conta com mais quatro, sendo um para menu, um para tela principal, outro para desfazer e por fim um para busca/ajuda.

As versões de hardware que não apresentam a barra de navegação tradicional com os quatro botões apresentam uma barra virtual posicionada no canto inferior da tela, contendo as mesmas funções.

A Figura 8 exemplifica dispositivos com barra de navegação física em (a) e em (b) dispositivos com barra de navegação virtual e supressão do botão de busca, sendo acrescentada a função na barra superior.

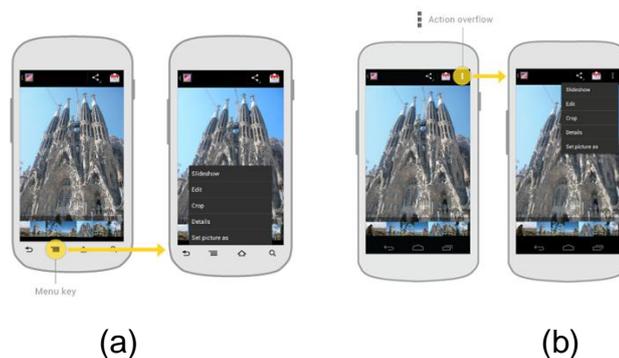


Figura 8 – Exemplos de smartphones com (a) barra de navegação física; (b) barra de navegação virtual

Fonte: (Google Inc., 2012)

Em se tratando de recursos de software, deve-se criar uma lista de requisitos básicos ao publicar o aplicativo na loja virtual, limitando o download aos dispositivos compatíveis. Caso o desenvolvedor opte por fazer uma única versão, deve-se deixar disponíveis as alterações cabíveis em um menu de configurações.

Outro ponto essencial é a compatibilidade do design do *app*. Devem ser planejadas interfaces que se adequem à orientação do dispositivo e ao tamanho da tela e sua resolução. Os conhecidos layouts fluidos garantem que independente da resolução e do tamanho da tela, os elementos que o compõem mantenham a proporcionalidade desejada, bem como sua posição na tela. Apesar de demandar um tempo maior na sua confecção, é o mais adequado para interfaces de aplicativos para dispositivos móveis.

2.2.5.2 Comunicação

A comunicação entre dispositivos pode ser estabelecida através de protocolos de comunicação previamente estabelecidos. Dentre os mais utilizados encontram-se WI-FI, 3G, Bluetooth e redes Ad-Hoc.

Para cada objetivo é necessário explorar o melhor protocolo. Alguns sistemas operacionais podem comunicar entre si usando as regras comuns de redes de computadores, entretanto, alguns métodos são privados. O Bluetooth, por exemplo, apesar de ser uma facilidade comum a todos os sistemas operacionais não possui comunicação aberta entre todos os SO's. Dispositivos com o sistema *iOS* só podem comunicar entre si, enquanto que os que possuem *Android* e *Symbian*, por exemplo, podem comunicar livremente entre si e com computadores pessoais.

Por se tratarem de aparelhos celulares, os *smartphones* possuem acesso à rede telefônica através de um cartão SIM para identificar, controlar e armazenar dados de telefones celulares de tecnologia GSM (Global System for Mobile Communications).

Pode-se tirar vantagem desse fato, explorando os recursos de envio de mensagens de texto, vídeo e voz através dos planos de telefonia adquiridos pelo usuário.

Também é possível acessar dados de contatos armazenados no cartão SIM para criação de bancos de dados, por exemplo, mantendo seguras as informações pessoais do usuário e garantindo que os dados pessoais serão os mesmos independentemente do aparelho utilizado.

2.3 Realidade Virtual e Aumentada

Realidade Virtual – RV - é uma interface avançada para aplicações computacionais, onde o usuário pode navegar e interagir, em tempo real, em um ambiente tridimensional gerado por computador, usando dispositivos multisensoriais (Kirner & Kirner, 2008).

A Realidade Virtual apresenta-se como uma ferramenta capaz de, ao ser utilizada como ferramenta de treinamento e aprendizagem, estimular a criatividade do aprendiz, contribuindo, significativamente para a aquisição de conhecimento de forma implícita, pois incentiva a aprendizagem processual (Bullinger, Rößler, & Müller-Spahn, 1996). Este tipo de aprendizagem processual inclui processos complexos que se caracterizam pelo fluxo inconsciente de artefatos mentais e de repetições freqüentes e regulares. É importante ressaltar que a RV apóia a aprendizagem processual de forma inconsciente, o que contrasta com os métodos pedagógicos convencionais, que apóiam, principalmente, à aquisição explícita de

conhecimento de forma controlável e consciente (Hofman, Hock, & Müller-Spahn, 1995).

A RV facilita e torna mais intuitivos os processos de abstração e exteriorização de informações, pois ela proporciona certa linearidade nestes processos [Bullinger, Müller-Spahn e Rößler, 1996], o que podem ser explicados a seguir. As pessoas ao interagirem no ambiente real por meio de translação, rotação e visualização em 3D criam um modelo mental, isso significa que elas abstraem conceitos e informações do ambiente real e as representam mentalmente em 3D.

A Figura 9, apresentada por (Bullinger, Rößler, & Müller-Spahn, 1996) mostra que a Realidade Virtual contempla, com uso de elementos tridimensionais, a visualização da informação e a interação do usuário com a mesma, enquanto que tecnologias CAD oferecem recursos de visualização 3D e interação 2D e os protótipos 2D que operam apenas com recursos 2D.

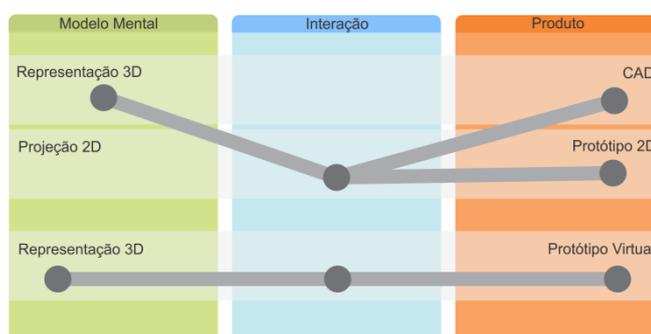


Figura 9 – Modelo Mental de produto.

Fonte: Adaptação de (Bullinger, Rößler, & Müller-Spahn, 1996)

Um dos principais diferenciais das tecnologias de RV é que seus ambientes são compostos de objetos virtuais que oferecem respostas comportamentais quando o usuário interage no cenário. A utilização de tais objetos para práticas educacionais deve ser acompanhado de perto por diretrizes pedagógicas específicas para que sejam considerados objetos de aprendizagem.

Por outro lado, Realidade Aumentada – RA – consiste na inserção de objetos virtuais no ambiente físico, mostrada ao usuário, em tempo real, com o apoio de algum dispositivo tecnológico, usando a interface do ambiente real, adaptada para visualizar e manipular os objetos reais e virtuais (Kirner & Kirner, 2008).

A Realidade Aumentada aplica-se em todos os sentidos humanos (Azuma, 2001) e proporciona ao usuário uma interação segura, sem necessidade de

treinamento, uma vez que ele pode trazer para o seu ambiente real objetos virtuais, incrementando e aumentando a visão que ele tem do mundo real (Kirner & Zorzal, 2005). Isto é obtido por meio de técnicas de Visão Computacional e de Computação Gráfica/Realidade Virtual, o que resulta na sobreposição de objetos virtuais com o ambiente real (Billinghurst, Kato, & Poupyrev, 2001).

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).

Para a visualização de um objeto virtual em uma cena real, pode-se utilizar um marcador fiducial, como exemplificado na Figura 10, que delimita a área da sobreposição dos objetos virtuais. Os marcadores são definidos como padrões a serem reconhecidos pelo sistema. Uma vez reconhecido, verifica-se na imagem capturada pela câmera, a uma taxa de quadros por segundo, a existência desse padrão. Quando encontrado, sua posição é utilizada para mapear o posicionamento dos elementos virtuais.

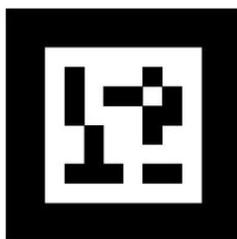


Figura 10 – Exemplo de marcador fiducial

Fonte: Acervo pessoal, 2012

A escolha do melhor marcador para cada aplicação deve ser feita com base nos seguintes parâmetros:

- Desenho do marcador
- Ambiente
- Roupas do usuário
- Iluminação

Com o intuito de aprimorar a utilização de RA, contornando os problemas causados pela má escolha de um marcador, podem ser utilizados marcadores mais complexos ou mesmo capturar características da própria cena e utilizá-las como padrões de posicionamento.

A escolha entre um sistema com ou sem marcador deve ser feita com base nos parâmetros de *hardware* utilizados, para que estes sejam aproveitados de forma eficiente e o resultado esperado seja atingido da melhor forma possível.

Com a sobreposição de objetos virtuais sobre a realidade, 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 (Sa & Churchill, 2012).

Com o desenvolvimento de dispositivos de comunicação móvel em âmbito de processamento, capacidade de armazenamento e melhorias de *hardware*, muitos engenheiros e designers têm pensado e criado novas experiências de interação com o usuário. A Realidade Aumentada é uma área que, associada à comunicação móvel, permite ao usuário experiências interativas e dinâmicas na apresentação de um conteúdo (Sa & Churchill, 2012).

CAPÍTULO 3

TRABALHOS CORRELATOS

3.1 Introdução

São apresentados a seguir trabalhos relacionados ao ensino, que fazem uso de diferentes tecnologias. São expostos também projetos de uso de m-learning no processo de aprendizagem de diferentes áreas, bem como o uso de Realidade Aumentada para fins didáticos.

3.2 MIT Magic Paper

O trabalho desenvolvido pelo Massachusetts Institute of Technology (MIT) em 2006 conhecido como *Magic Paper* consiste em um sistema de criação de modelos bidimensionais com interações físicas (Davis, 2007).

O sistema foi desenvolvido para uso em computadores pessoais com mouses ou mesas digitalizadoras como hardware de entrada ou para dispositivos com telas capazes de reconhecer o toque, como lousas digitais.

O *Magic Paper* recebe como dados de entrada, esboços de objetos em duas dimensões e os transforma em figuras geométricas com características físicas. Estas podem ser inseridas desenhando códigos e símbolos que as representem. Uma seta para baixo, por exemplo, simboliza a ação da gravidade sobre o ambiente.

A Figura 11 apresenta em (a) a etapa de criação do ambiente e em (b) a execução das leis físicas sobre os elementos criados.

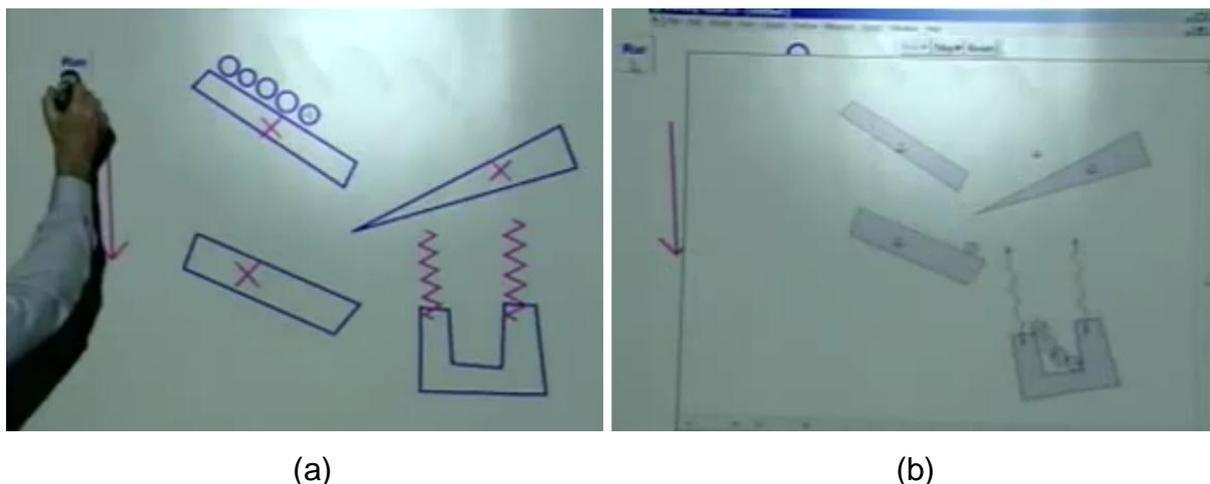


Figura 11 – Magic Paper. (a) Desenho do sistema físico; (b) Animação do modelo.

Fonte: (Davis, 2007)

São explorados elementos como blocos, molas, cilindros e bolas (representados por circunferências), que podem ser conectados ou dispostos isoladamente no cenário. O controle da animação é feito através de barras de ferramentas contidas no software e contam com técnicas que facilitam a compreensão das mesmas, garantindo a melhor visualização do espaço.

O *Magic Paper* apresenta algoritmos de correção do traçado, aproximando o esboço ao elemento geométrico bidimensional. Apesar da vasta biblioteca de elementos físicos e de forças de interação, a interação com o usuário é limitada ao uso do mouse ou do simples toque da caneta, não apresentando interações gestuais.

O ambiente criado limita-se ao tamanho fixo do quadro, não permitindo a expansão do mesmo através de técnicas como alteração da escala dos elementos.

O *Magic Paper* foi desenvolvido em linguagem C# e sua função se limita ao reconhecimento de formas esboçadas. A animação e interação física é feita pelo aplicativo *Working Model 2D*, desenvolvido pela *DST Design Simulation Technologies* (DST, Design Simulation Technologies, 2012).

O *Working Model 2D* fornece soluções precisas para problemas complexos de engenharia de simulação de movimento mediante poderosa simulação e análise 2D de movimentos cinemáticos e dinâmicos baseados em Física.

O formato de entrada de dados do software é o DXF. O *Magic Paper* é responsável por criar os modelos físicos e formata-los de forma que o *Working*

Model 2D realize as animações pertinentes. Possui linguagem *script* completa estilo “*Visual Basic*”, com depurador interno.

Contando com uma biblioteca de *scripts* predefinidos, o software permite ao usuário simular várias propriedades e recursos estruturais, bem como controlar a simulação e visualizar informações matemáticas sobre seus diferentes estados.

A interação natural é uma nova forma de interação com o software, tornando possível descrever objetos e ações por meio de esboços e gestos de maneira natural. Uma vez que um esboço é reconhecido, a informação que ele contém pode ser utilizada para outros aplicativos de simulação. A Microsoft incluiu os recursos criados pelo *Magic Paper* em seus *Tablet PCs*, descrevendo sua funcionalidade como “uma simulação de movimento para o *Tablet PC* que demonstra Física usando objetos desenhados pelo usuário” (Microsoft Research, 2012).

A Microsoft, em parceria com o MIT, criou o *iCampus*, uma colaboração de pesquisas entre a Microsoft Research e o instituto, introduzindo o *Natural Interaction*, baseado na tecnologia desenvolvida para o *Magic Paper*.

São características do *Natural Interaction*:

- Foi implementada para vários domínios do conhecimento, incluindo dispositivos mecânicos 2D e diagramas de software UML;
- O ruído inerente ao desenho é tratado nos esboços à mão livre;
- Usa uma grande base de dados para resolver ambiguidades;
- Provê uma interação modal para desenho e edição.

A portabilidade do *Magic Paper* para sistemas operacionais móveis é uma proposta de ação, visando o apoio ao ensino. É importante ressaltar que apenas sua utilização em dispositivos móveis não caracteriza a *m-learning*, mas seu uso assistido por professores e tutores presenciais e/ou virtuais pode ser considerado uma prática de tal modalidade.

3.3 In-Place 3D

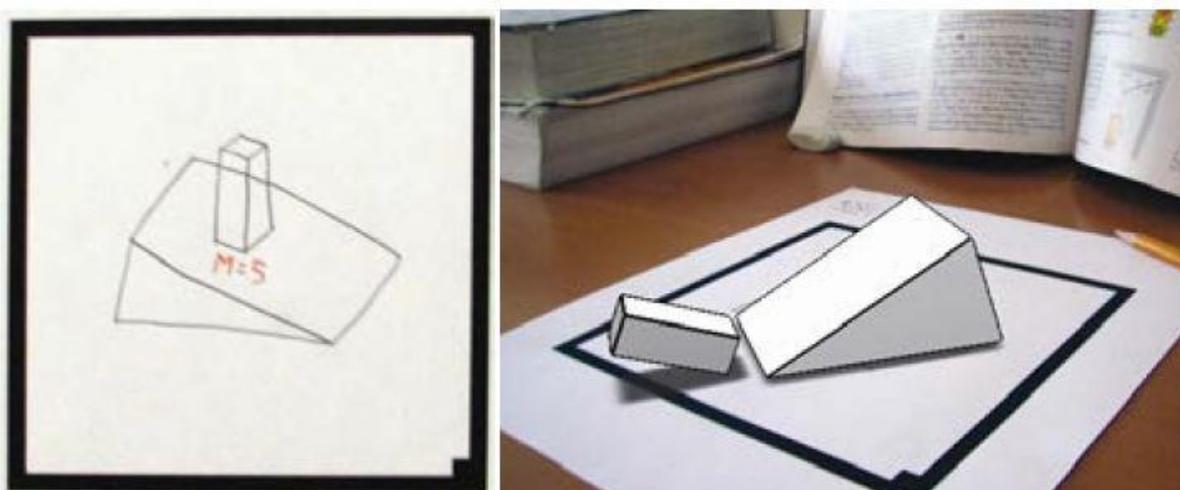
O *In-Place 3D* é um framework para autoria de cenas tridimensionais para realidade aumentada baseado em desenhos à mão livre. Com este framework, é possível transformar esboços em sistemas em três dimensões com a possibilidade de interação, permitindo ao usuário controlar as etapas da animação, bem como alterar as propriedades físicas dos elementos contidos no ambiente (Bergig, Hagbi, El-

Sana, & Billinghamurst, *In-Place 3D Sketching for Authoring and Augmenting Mechanical Systems*, 2009).

Desenvolvido pelo *Hit Lab New Zealand* em 2009, o *In-Place 3D* faz uso de técnicas de processamento de imagens e realidade aumentada, bibliotecas de desenvolvimento de ambientes e interações físicas.

Os esboços são feitos com base no desenho em perspectiva ortogonal. São usadas técnicas de processamento de imagens e reconhecimento de padrões para reconhecimento dos símbolos desenhados e capturados por uma câmera comum.

A Figura 12 apresenta em (a) a autoria de um sistema mecânico, criado sobre um marcador especial que delimita o espaço da cena. Em (b) tem-se o ambiente virtual gerado e inserido no ambiente real sobre o marcador.



(a)

(b)

Figura 12 - InPlace 3D (a) Autoria do sistema mecânico, esboçado no papel; (b) Ambiente de Realidade Aumentada construído com base no modelo desenhado.

Fonte: (Bergig, Hagbi, El-Sana, & Billinghamurst, In-Place 3D Sketching for Authoring and Augmenting Mechanical Systems, 2009)

O tratamento das imagens permite a criação de elementos sólidos baseados em triedros e tetraedros e forças interativas como atrito, gravidade e velocidade. A criação do modelo é gerada e mostrada no vídeo sobreposta à imagem da câmera. Os movimentos e transformações dos objetos são feitos com a adição de marcadores de controles à cena e a movimentação da câmera.

O framework de desenho é genérico e, portanto, pode ser utilizado para diferentes casos de uso. Sua utilização gerou diferentes publicações em áreas de ensino e criação de mundos virtuais.

A aplicação inicia seu processo capturando o vídeo da *webcam* do usuário e mostrando seu conteúdo na tela. Um quadro preto é usado para delimitar a área de desenho. Uma vez que o quadro é encontrado e posicionado adequadamente, dá-se início à procura pelo esboço do sistema mecânico. Cada rascunho pode conter triedros (com preenchimento automático de linhas ocluídas) desenhados em cor preta e projeção ortográfica. Ademais, aceitam-se anotações em vermelho.

O usuário pode desenhar o sistema passo-a-passo ou visualizar um sistema completo. Os modelos podem ser definidos como estáticos ou dinâmicos e interagem entre si em um sistema completo. Os elementos que se sobrepõem devem ser desenhados em cores diferentes ou introduzidos em etapas diferentes.

O movimento da câmera determina os parâmetros da transformação de um objeto. A interação com a cena pode basicamente constituir-se de escala, translação e rotação dos elementos que a compõem, o que pode ser feito com a alteração do esboço no papel.

É possível também alterar um esboço pré-definido acrescentando a ele elementos novos. Pode-se, por exemplo, usar um papel transparente sobre uma página de um livro para visualizar diferentes formas de interação.

A Figura 13 apresenta a estrutura do software *In-Place 3D* segundo (Bergig, Hagbi, El-Sana, & Billinghamurst, *In-place 3D sketching for authoring and augmenting mechanical systems*, 2009).

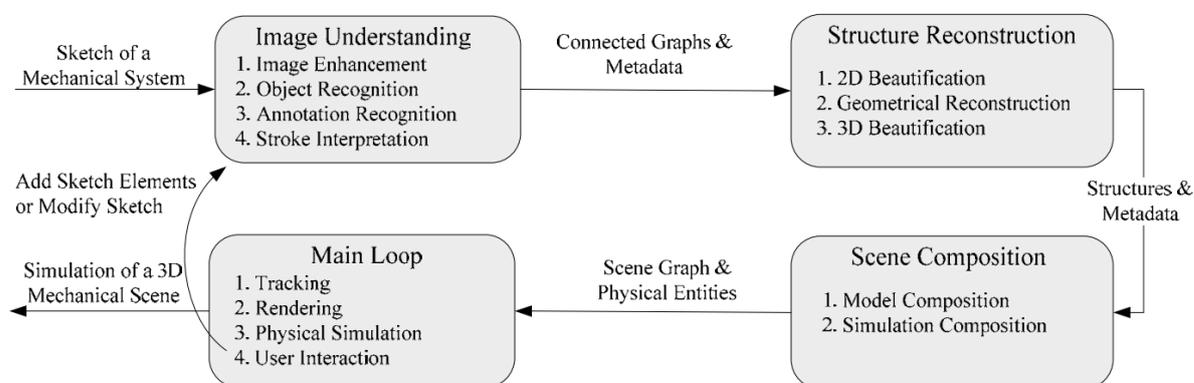


Figura 13 – Estrutura do software *In-Place 3D*

Fonte: (Bergig, Hagbi, El-Sana, & Billinghamurst, *In-Place 3D Sketching for Authoring and Augmenting Mechanical Systems*, 2009)

O processo de reconhecimento de padrões é feito em quatro etapas principais:

1. Processamento Digital de Imagens
2. Reconstrução da Estrutura
3. Composição da Cena
4. Loop Principal

O processamento da imagem consiste em reforçar a imagem desenhada, reconhecer o objeto esboçado, diferenciar e reconhecer as anotações de propriedades físicas e geométricas e interpolação do contorno da forma, garantindo o tratamento de oclusão de arestas.

A reconstrução da estrutura é responsável pela suavização da forma bidimensional (eliminar traços redundantes), reconstrução geométrica e a criação do modelo tridimensional.

A composição da cena atribui a cada modelo tridimensional suas propriedades físicas relativas e realiza a simulação das leis físicas.

Por fim, tem-se o loop principal, responsável pelo seguimento do marcador, renderização da cena criada, simulação dos eventos físicos e interação com o usuário. Este processo se repete constantemente, atualizando os dados do sistema em uma determinada taxa de quadros por segundo.

A cena virtual não se limita ao marcador, apenas o utiliza como orientador no espaço. Entretanto, assim como o *Magic Paper*, o *InPlace 3D* tem espaço limitado para criação do modelo mecânico.

A interação com o ambiente, realizada através da câmera, é melhor utilizada em um ambiente controlado, com uma câmera fixa. Para dispositivos móveis, esta forma de interação deve ser repensada para que aborde os conceitos de interação gestual, utilizando o toque do usuário e a orientação do dispositivo.

3.4 ILoveSketch

Trata-se de um software desenvolvido por Seok-Hyung bae, Ravin Balakrishnan e Karan Singh, capaz criar ambientes tridimensionais baseados em esboços feitos em dispositivos de captura por toque (Bae, Balakrishnan, & Singh, ILoveSketch: As-natural-as-possible system for creating 3D curve models, 2008).

Este sistema pode ser usado com navegação 2D ou 3D e cria curvas NURBS. Foi desenvolvido para profissionais do design, possibilitando a interação do usuário com seus modelos 3D com gestos comuns, existentes em uma biblioteca coesa de gestos.

A aplicação diferenciada do software é a navegação e criação de elementos usando um ambiente tridimensional. Desenho e transformações dos mesmos se misturam em um ambiente sólido e consistente.

São trabalhados gestos como ponto, curva, laço, loop, entre outros, para desenho, transformação e movimentação do ambiente.

Todos os traços do usuário são tratados a fim de melhorar o desenho e torná-lo mais uniforme com os padrões de projeto. São evitadas as redundâncias de traços, para que não haja ambiguidade no processamento dos dados, deixando os contornos definidos.

O ambiente conta com as principais ferramentas de transformação dos modelos, que alteram suas propriedades geométricas e estéticas. Assim como os grandes editores de modelos 3D, o ILoveSketch propõe diferentes câmeras e perspectivas de vistas.

Observa-se na Figura 14 a utilização do sistema e alguns exemplos de resultados.

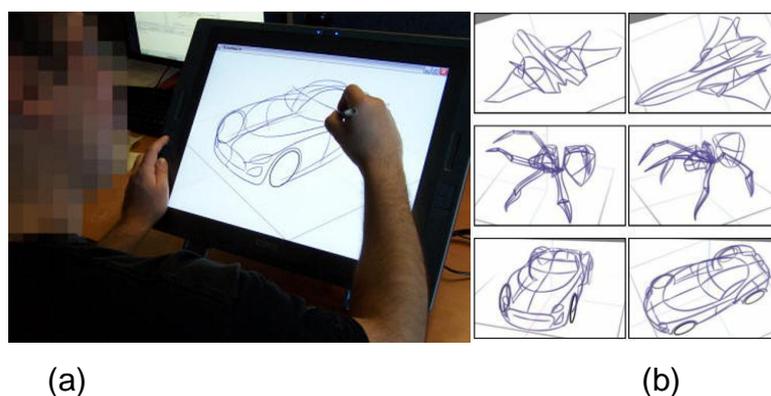


Figura 14 - ILoveSketch (a) Criação do esboço; (b) Trabalhos finais gerados.

Fonte: Adaptado de (Bae, Balakrishnan, & Singh, *ILoveSketch: As-natural-as-possible system for creating 3D curve models*, 2008)

A idéia principal é rever os métodos tradicionais de desenho em perspectiva, a fim de aprimorar um sistema de esboços tridimensionais e determinar os aspectos que podem ser automatizados ou são desnecessários para um sistema que permite

trocas frequentes de visualizações 3D (Bae, Balakrishnan, & Singh, EverybodyLovesSketch: 3D Sketching for a Broader Audience, 2009).

O sistema ILoveSketch foi concebido e implementado pela integração de muitas técnicas, sendo algumas delas inovadoras, outras já existentes e adaptadas. O sistema foi avaliado por profissionais de Design Gráfico e comprovado com potencialmente útil como um produto de mercado.

Existem alguns pontos a serem estudados sobre o sistema desenvolvido, como:

- Um estudo formal do desempenho de cada ponto de vista;
- Combinação da caneta de desenho com gestos de toque;
- Uso de gestos de toque para navegação da câmera

Apesar das boas ferramentas de desenho, o software não conta com ferramentas para animação e interação maior que a mudança de pontos de vista. O espaço de criação pode ser aumentado, através de gestos, mas não há interação entre os elementos criados. Estes se comportam como elementos estáticos.

3.5 Resumo Comparativo dos Trabalhos Analisados

Apesar de apresentarem escopos diferentes, os trabalhos relacionados acima contribuem para a elaboração desta dissertação em diferentes formas. O trabalho de maior inspiração é o In-Place 3D, que mais se preocupa em transformar um esboço 2D em um ambiente 3D. Observa-se, entretanto, que o rascunho é gerado pelo usuário de forma tradicional, usando lápis e papel.

Pensando em utilizar outros recursos de entrada de dados, o *Magic Paper* apresenta um método interessante, utilizando uma lousa digital, ou mesmo periféricos comuns de um computador *desktop*, como o mouse, ou uma mesa digitalizadora.

Ao se pensar na criação do ambiente 3D, o ILoveSketch é bastante intuitivo e trabalha com gestos (mesmo que sem entrada de toque) para alternar entre as vistas e perspectivas do modelo.

A

Tabela 2 apresenta os tópicos abordados em cada trabalho, comparando-os entre si.

Tabela 2 – Quadro comparativo entre os trabalhos correlatos

Tópico Abordado	<i>Magic Paper</i>	<i>In-Place 3D</i>	<i>ILoveSketch</i>
Esboços 2D	✓	✓	✓
RV	✗	✗	✓
RA	✗	✓	✗
Interação por gestos	✗	✗	✓
Compatibilidade com dispositivos 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, seus casos de uso detalhados e diagramas de classe com as principais classes envolvidas.

4.2 Requisitos Funcionais

Os principais requisitos funcionais de um sistema de reconhecimento de padrões envolvem a correta aquisição dos dados, o processamento dos mesmos e a apresentação da simplificação dos dados obtidos.

A eficácia do reconhecimento de padrões gestuais não envolve apenas a boa aquisição das coordenadas de cada trecho do caminho percorrido, mas também dos seguintes parâmetros:

- Tempo gasto na aquisição;
- Número de pontos que compõem o caminho percorrido;
- Tipo de resposta esperada;

Para que a coleta de tais dados seja feita de maneira correta, é preciso levar em consideração as características físicas do dispositivo móvel, como o tamanho e a resolução da tela, bem como o tipo de display adotado (resistivo ou capacitivo).

Para que os dados possam ser utilizados em diferentes dispositivos, é importante estabelecer uma medida unitária padrão a ser utilizada em qualquer caso de uso.

O reconhecimento dos padrões de entrada requer o bom uso do processamento do aparelho. Para isso, deve-se pensar nas melhores formas de armazenamento dos dados e a comunicação entre eles. Boas práticas de programação podem garantir o melhor uso dos recursos disponíveis.

Um dicionário coeso de gestos deve ser feito usando a mesma metodologia de captura de dados para comparação. Desta maneira, a probabilidade de erro por comparação diminui, garantindo a coerência dos resultados.

Provavelmente a etapa do processo que requer maior atenção, a transição entre os esboços bidimensionais e o ambiente virtual necessita de uma série de parâmetros que devem ser bem ajustados/calculados.

Entre eles, destacam-se as propriedades físicas do objeto bidimensional:

- Coordenadas do objeto no plano;
- Altura e largura do esboço;
- Escala e rotação do padrão
- Método de extrusão e seus parâmetros.

Entre os parâmetros físicos de cada elemento transformado, destacam-se:

- Peso do elemento;
- Aceleração, velocidade e outras forças aplicadas ao objeto;
- Atritos envolvidos;

Semelhante à criação do ambiente virtual, a criação do ambiente aumentado deve levar em consideração as propriedades físicas do modelo gerado, garantindo a interação dos objetos tridimensionais presentes na cena.

Devem ser observadas também características do ambiente real para que a experiência seja realizada da melhor forma possível. São analisados os seguintes atributos:

- Iluminação
- Dispositivo de *hardware* utilizado
- Reflexão da luz na superfície na qual se localiza o marcador

4.3 Requisitos Não Funcionais

Os requisitos não funcionais do sistema são relacionados ao uso do aplicativo em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenibilidade e tecnologias envolvidas. Podem constituir em restrições aos requisitos funcionais.

O sistema deverá ser desenvolvido para a plataforma *Android*, em linguagem *Actionscript* 3.0. A consulta ao Banco de Dados deve ser feita utilizando linguagens *PHP* e *MySQL*.

O aplicativo deverá comunicar-se entre usuários e com o banco de dados. Para tal, deve ser estabelecida uma conexão com a Internet.

A atualização da tela de bate-papo deve ser feita a cada cinco segundos para garantir a melhor comunicação entre os usuários. Ao inicializar o aplicativo, verificar a conexão com a Internet.

Usuários deverão operar o aplicativo após treinamento realizado por meio de vídeo explicativo.

O aplicativo deve ser compatível a dispositivos com o sistema *Android* com versão 2.2 ou superior e suporte ao Adobe AIR, instalável automaticamente caso não esteja instalado antes.

4.4 Casos de Uso

Nesta sessão serão apresentados os principais casos de usos do aplicativo desenvolvido, desde a inserção de novos usuários, passando pelo registro dos mesmos, a criação de esboços bidimensionais, a transição para os ambientes virtual e aumentado e o compartilhamento de informações.

Observa-se que a interação do usuário com o sistema baseia-se na inserção e alteração de dados do perfil do usuário, na criação dos modelos tridimensionais e na comunicação entre usuários de um mesmo grupo. Os casos de uso distribuem-se nessas grandes áreas, conforme apresentado na Figura 15.

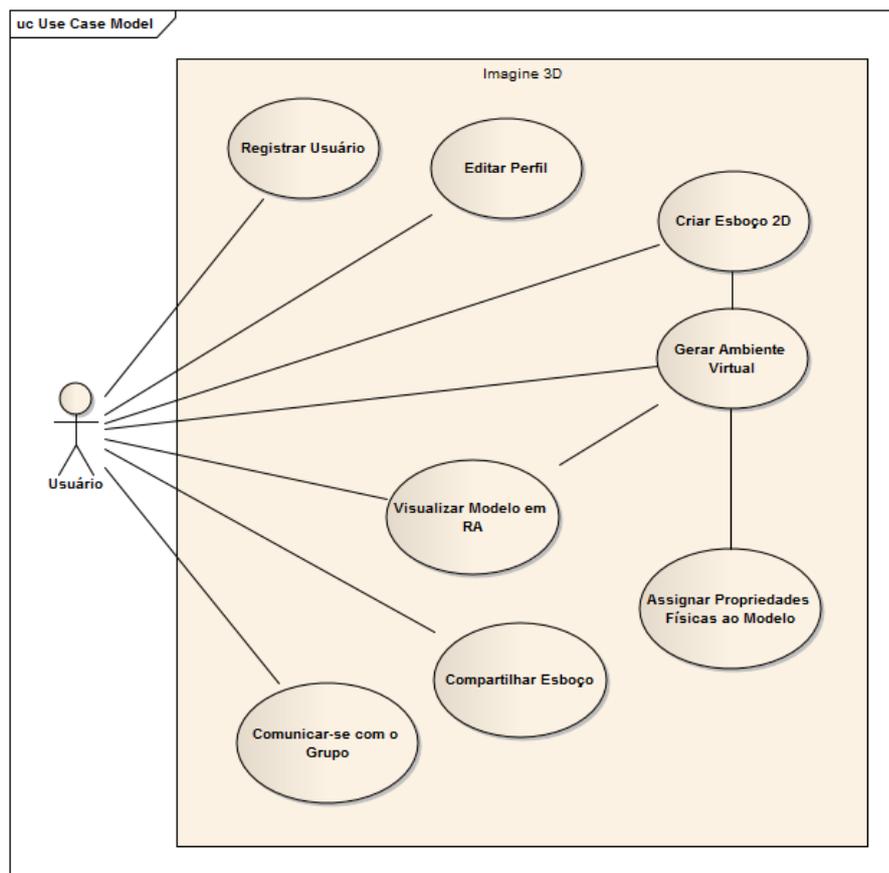


Figura 15 – Diagrama de Casos de Uso

Fonte: Acervo pessoal, 2012

É importante destacar a dependência entre alguns casos de uso do sistema como, por exemplo, o caso “Gerar Ambiente Virtual” depende do caso “Criar Esboço 2D” por uma relação de uso simples, isto é, uma vez que este é finalizado, aquele pode ser realizado.

São descritos a seguir os atores e os casos de uso do sistema, enfatizando suas funções para o funcionamento geral do aplicativo.

4.4.1 Descrição dos Atores

Os atores do sistema, bem como suas atribuições podem ser vistos na **Error! Reference source not found.** Optou-se por não separar os usuários “aluno” e “professor” em grupos diferentes, uma vez que a primeira versão do aplicativo não permite ao professor privilégios diferentes. Para simplificar o sistema, adotou-se a notação “usuário” para todo o grupo.

Quadro 1 - Atores do sistema

Nome	Descrição	Atribuições
Usuário	Aluno ou professor utilizadores do sistema	Cadastrar suas informações em uma base de dados comum. Gerar e compartilhar modelos virtuais. Comunicar-se com o grupo de usuários envolvidos na atividade.

Para as próximas versões do sistema, estima-se a separação dos perfis, concedendo ao professor privilégios de moderador do grupo, permitindo a ele selecionar e excluir alunos de um grupo, por exemplo.

4.4.2 Descrição dos Casos de Uso

São descritos nos Quadros 2 a 9 os casos de uso apresentados anteriormente.

Quadro 2 - Descrição do caso de uso "Registrar Usuário"

Caso de Uso	<i>Registrar Usuário</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Criar um novo usuário, inserindo seus dados de contato e outras informações pessoais.</i>
Pré-Condição	-
Fluxo Normal	<ol style="list-style-type: none"> 1. Acessar a tela de cadastro. 2. Inserir nome completo; 3. Inserir apelido; 4. Criar senha de acesso; 5. Informar email de contato; 6. Cadastrar foto.
Fluxos Excepcionais	-
Pós-Condição	<i>Usuário Inserido com sucesso</i>

Quadro 3 - Descrição do Caso de Uso “Editar Perfil”

Caso de Uso	<i>Editar Perfil</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Editar os dados de contato e informações pessoais de um usuário previamente cadastrado.</i>
Pré-Condição	<i>Estar registrado no sistema</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela de cadastro.</i> 2. <i>Selecionar a opção “Editar Perfil”</i> 3. <i>Inserir dados de acesso;</i> 4. <i>Alterar dados cadastrais:</i> <ol style="list-style-type: none"> a. <i>Inserir nome completo;</i> b. <i>Inserir apelido;</i> c. <i>Criar senha de acesso;</i> d. <i>Informar email de contato;</i> e. <i>Cadastrar foto.</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Dados atualizados com sucesso</i>

Quadro 4 - Descrição do Caso de Uso “Criar Esboço 2D”

Caso de Uso	<i>Criar Esboço 2D</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Por meio de gestos, desenhar formas geométricas que correspondem ao sistema físico desejado.</i>
Pré-Condição	<i>Estar registrado no sistema</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “Desenho”</i> 2. <i>Desenhar uma forma geométrica utilizando gestos e toque;</i> 3. <i>Escolher entre a forma simplificada sugerida ou a forma inserida;</i> 4. <i>Posicionar a figura na tela usando gestos de arrastar e soltar</i> 5. <i>Repetir os passos 1 a 3 para todas as formas do sistema;</i> 6. <i>Salvar modelo criado.</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Modelo criado e armazenado localmente.</i>

Quadro 5 - Descrição do Caso de Uso “Compartilhar Esboço”

Caso de Uso	<i>Compartilhar Esboço</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Enviar o esboço bidimensional criado ao banco de dados comum</i>
Pré-Condição	<i>Estar registrado no sistema e ter criado o esboço 2D</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “CHAT”;</i> 2. <i>Selecionar o modelo a ser compartilhado;</i> 3. <i>Enviar a notificação de compartilhamento.</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Esboço compartilhado com os usuários do grupo</i>

Quadro 6 - Descrição do Caso de Uso “Gerar Ambiente Virtual”

Caso de Uso	<i>Gerar Ambiente Virtual</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>A partir do esboço 2D criado, gerar um ambiente virtual correspondente.</i>
Pré-Condição	<i>Estar registrado no sistema e ter criado o esboço 2D</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “Desenho”;</i> 2. <i>Pressionar sobre a forma para abrir a tela de edições;</i> 3. <i>Definir o método de extrusão adotado;</i> 4. <i>Ajustar parâmetros de rotação e escala;</i> 5. <i>Repetir os passos 1 a 3 para cada forma geométrica inserida;</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Ambiente virtual gerado e salvo localmente</i>

Quadro 7 - Descrição do Caso de Uso “Assignar Propriedades Físicas ao Modelo”

Caso de Uso	<i>Assignar Propriedades Físicas ao Modelo</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Uma vez gerado o modelo virtual, assignar as propriedades físicas de cada elemento.</i>
Pré-Condição	<i>Estar registrado no sistema, ter criado o esboço 2D e gerado o ambiente virtual.</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “Ambiente Virtual”</i> 2. <i>Pressionar sobre o elemento para abrir a tela de edições;</i> 3. <i>Definir as propriedades físicas do objeto;</i> 4. <i>Ajustar posição do objeto no espaço;</i> 5. <i>Repetir os passos 1 a 3 para cada elemento virtual;</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Propriedades assignadas com sucesso.</i>

Quadro 8 - Descrição do Caso de Uso “Visualizar Modelo em RA”

Caso de Uso	<i>Visualizar Modelo em RA</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Uma vez gerado o modelo virtual, visualizá-lo em Realidade Aumentada.</i>
Pré-Condição	<i>Estar registrado no sistema, ter criado o esboço 2D e gerado o ambiente virtual.</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “Realidade Aumentada”;</i> 2. <i>Posicionar o marcador em frente à câmera;</i> 3. <i>Reproduzir Animação.</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Reprodução do ambiente em RA</i>

Quadro 9 - Descrição do Caso de Uso “Comunicar-se com o Grupo”

Caso de Uso	<i>Comunicar-se com o Grupo</i>
Ator Principal	<i>Usuário</i>
Descrição	<i>Enviar ao grupo uma dúvida, sugestão ou comentário acerca do modelo criado ou o tema discutido.</i>
Pré-Condição	<i>Estar registrado no sistema.</i>
Fluxo Normal	<ol style="list-style-type: none"> 1. <i>Acessar a tela “Chat”</i> 2. <i>Inserir texto a ser enviado;</i> 3. <i>Submeter texto.</i>
Fluxos Excepcionais	-
Pós-Condição	<i>Mensagem enviada.</i>

4.5 Diagrama de Classes

Nesta sessão será apresentado o diagrama de classes do sistema. As classes foram criadas com base nos diagramas apresentados anteriormente, a fim de satisfazer os requisitos do sistema.

A Figura 16 apresenta as principais classes utilizadas no sistema, bem como seus atributos e principais funções.

O usuário do sistema é capaz de realizar três ações principais: comunicar-se com o grupo, criar e editar modelos e gerar e editar modelos tridimensionais. Cada esboço 2D é composto por formas que são criadas por meio de gestos. Os modelos 3D são compostos de objetos 3D simples e/ou criados por meio de extrusões. Por fim, os modelos 3D são submetidos ao Misturador de RA, que gera a simulação em realidade aumentada, apresentada em um marcador fiducial.

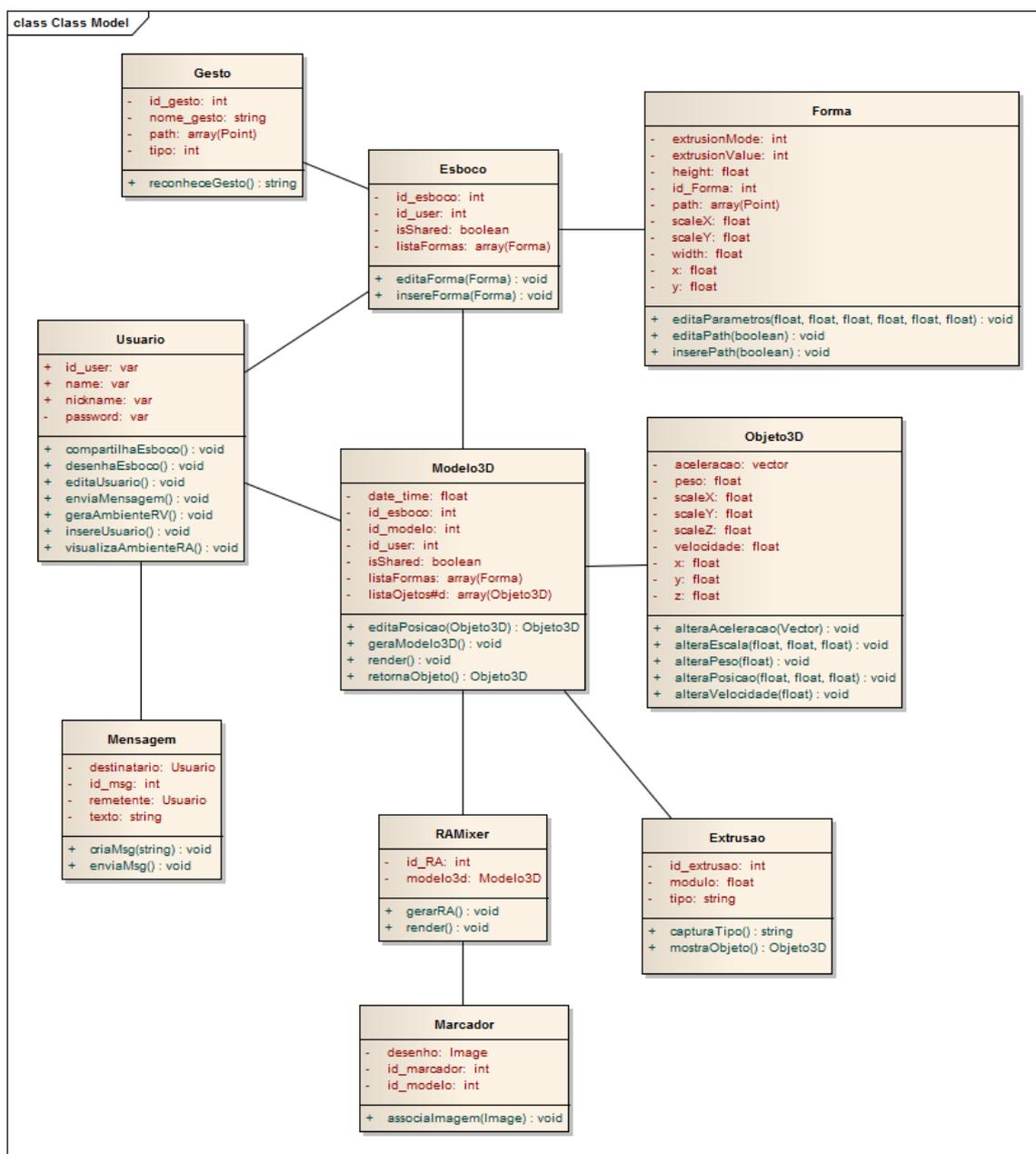


Figura 16 – Diagrama de classes do sistema

Fonte: Acervo pessoal, 2012

4.6 Pipeline

As cinco principais ações realizadas pelo usuário, apresentadas nas sessões 4.4 e 4.5 podem ser vistas na Figura 17

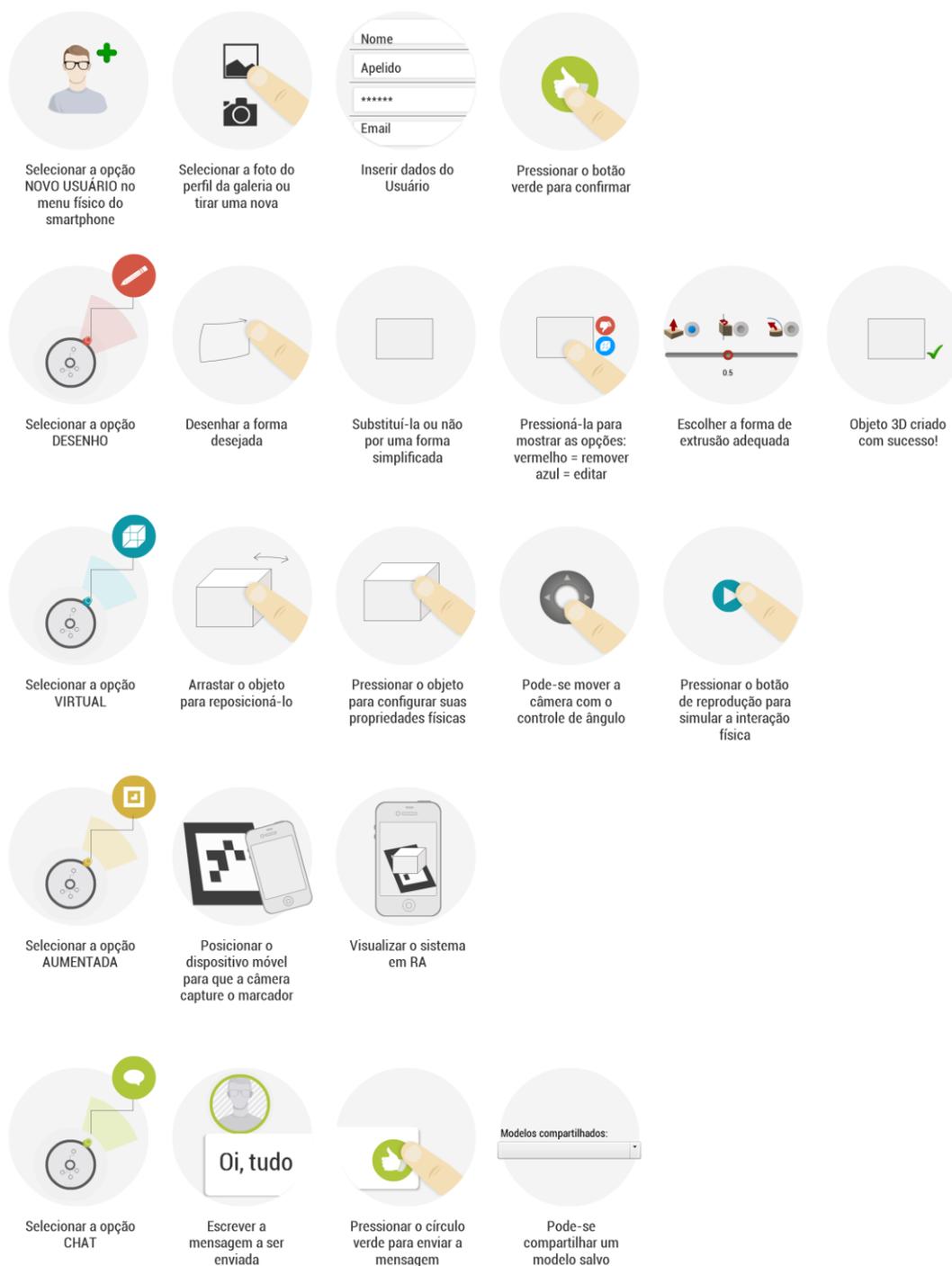


Figura 17 – Pipeline de utilizações do sistema

Fonte: Acervo pessoal, 2012

CAPÍTULO 5

DETALHES DA IMPLEMENTAÇÃO

5.1 Introdução

Este capítulo apresenta detalhes da implementação do aplicativo modelo, incluindo o detalhamento dos algoritmos de criação de esboços bidimensionais, sua transformação em elementos 3D, a comunicação entre dispositivos em uma rede, o design da interface e a navegação no aplicativo.

5.2 Estrutura do Aplicativo

O aplicativo gerado para estudo de caso apresenta uma arquitetura modular simples, na qual cada módulo é responsável por uma parte do processo e é interligado aos demais módulos. A Figura 18 apresenta o diagrama estrutural dos módulos do sistema.

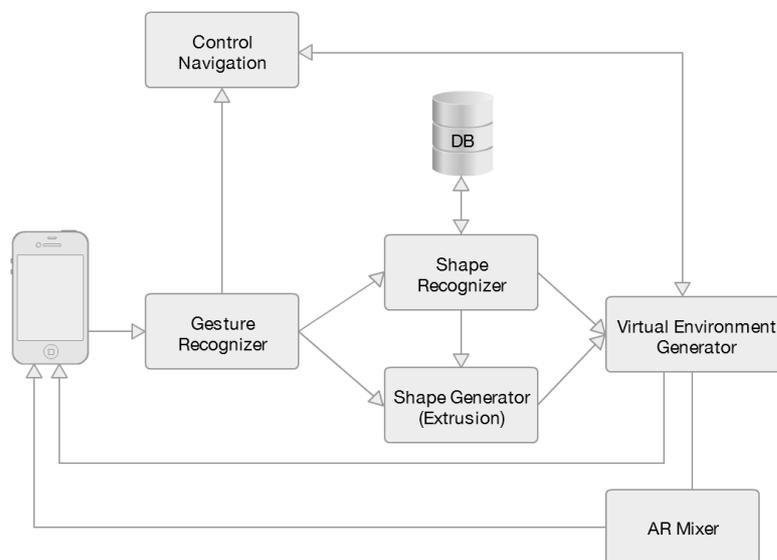


Figura 18 – Estrutura do aplicativo proposto.

Fonte: Acervo pessoal, 2012

Ao receber um padrão de gesto de entrada, o sistema passa pelo módulo **Gesture Recognizer**, que verifica se o gesto inserido é responsável por controle da navegação no *app* ou se ele faz parte da criação do ambiente virtual. Para este

aplicativo, foram adotados gestos com um dedo para desenhos de esboços e gestos com dois dedos para navegação e alteração de esboços.

O módulo **Control Navigation** realiza todas as operações de transição entre as telas do aplicativo e também a navegação no ambiente virtual criado.

Uma vez que os gestos são identificados como gestos de criação é necessário buscar um padrão existente no banco de dados do sistema. Esta busca é feita pelo módulo **Shape Recognizer**. Os algoritmos de busca utilizados serão descritos proximamente. Caso o padrão seja encontrado ele é desenhado na tela do usuário com um processo de simplificação de sua forma, eliminando traços desnecessários ou mesmo ruídos gerados na inserção do padrão.

Caso o gesto não seja encontrado no banco de dados, é necessário criar um novo padrão através da entrada do usuário. Esta função é desempenhada pelo **Shape Generator**, que além de desenhar uma nova forma bidimensional a transforma em um elemento 3D, segundo algoritmos desenvolvidos neste trabalho e apresentados nos próximos tópicos.

O módulo **Virtual Environment Generator** é responsável pela integração dos elementos 3D criados nos passos anteriores e a aplicação das forças físicas pertinentes a cada caso de estudo do usuário. O módulo então finaliza sua atuação apresentando o ambiente ao usuário ou encaminhando-o para a geração de Realidade Aumentada.

Finalizando o processo, o módulo **AR Mixer** cria o ambiente de Realidade Aumentada a partir do ambiente de RV criado anteriormente. Para isso, faz uso dos dispositivos de vídeo existentes no aparelho móvel, misturando os ambientes real e virtual.

Apesar de serem desenvolvidos em separado, cada módulo desempenha suas funções com qualidade de acordo com os seus dados de entrada. Desta forma, a conexão dos módulos deve ser pensada de forma a manter o melhor fluxo da informação.

O pipeline estrutural do aplicativo pode ser visto na Figura 19.

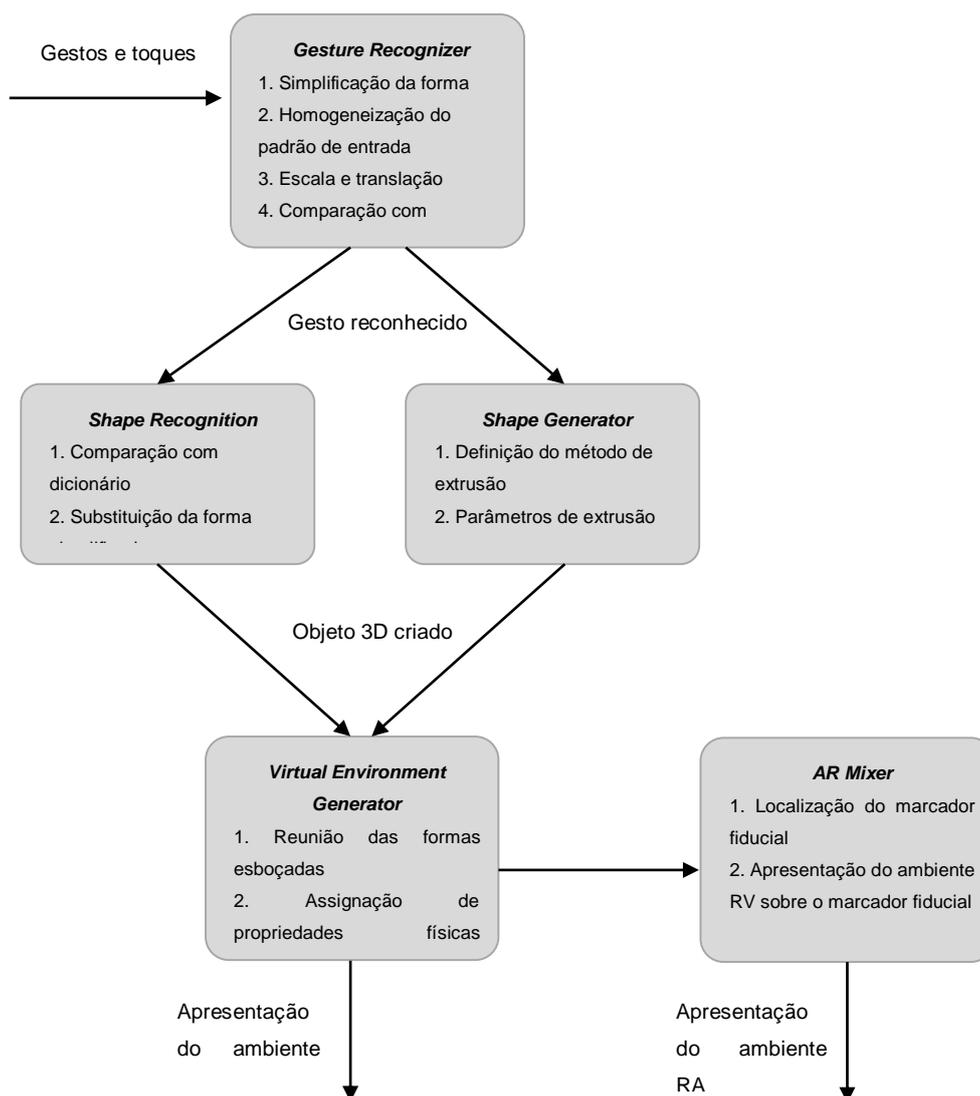


Figura 19 – Pipeline estrutural do aplicativo Imagine 3D

Fonte: Acervo pessoal, 2012

O procedimento para reconhecimento de um gesto e sua transformação em um objeto tridimensional começa com a entrada de dados, realizada pelo usuário por meio da interação de toque.

O módulo *Gesture Recognizer* é responsável pela transformação dos dados de entrada (conjunto de pontos que descrevem o caminho de toque) em uma descrição codificada do gesto reconhecido. Para isso são realizados métodos de homogeneização da forma e seu tratamento matemático, para que seja gerado um padrão universal.

O padrão codificado é então comparado com um dicionário de formas existentes. Caso ocorra uma correspondência, um padrão simplificado é sugerido ao

usuário. Caso contrário, o módulo *Shape Generator* é responsável pela inserção do modelo desejado.

Uma vez definido o modelo 3D desejado, o usuário deve inserir as propriedades físicas do elemento e pode assim, visualizar o ambiente virtual. Caso opte por uma experiência diferente, o usuário pode acionar o *AR Mixer*, responsável pela geração da Realidade Aumentada.

5.3 Estrutura de Navegação

Foram criadas cinco telas principais (*top level views*) arranjadas em um padrão no qual cada tela comunica com as demais em todos os passos do processo. Também foram criadas duas telas de edição para as configurações do *app*. Estas se comunicam apenas com a tela principal.

Durante o processo de criação de um esboço e sua transformação em ambiente virtual e/ou aumentado, são transmitidos dados entre uma tela e outra de modo local. A Figura 20 esquematiza a estrutura de telas do aplicativo.

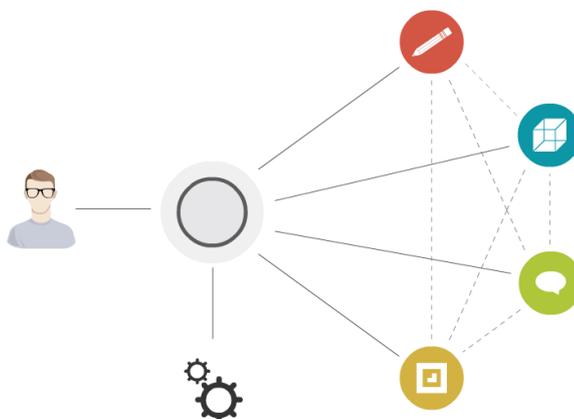


Figura 20 – Estrutura de telas do aplicativo modelo.

Fonte: Acervo pessoal, 2012

A navegação entre telas se dá por um único menu principal que apresenta três versões: exposto, contraído e aberto. Quando exposto, o menu em sua forma principal, apresenta todos as opções de telas do *app*, dando ao usuário liberdade em dispor as opções na tela da maneira que mais lhe agrada.

A versão contraída do menu aparece quando as telas principais estão abertas. Localizado no canto superior esquerdo, o menu apresenta parte da sua forma central e, em evidência, o ítem correspondente à tela aberta. Para alternar entre as telas, basta um leve toque sobre o menu para que ele abra seu leque de opções e aguarde a decisão do usuário.

A Figura 21 mostra o comportamento do menu principal em suas três versões básicas. A disposição dos itens do menu exposto é primeiramente estabelecida como mostrado em (a). Contudo, o usuário pode alterá-la para que fique de acordo com seu gosto e funcionalidade.

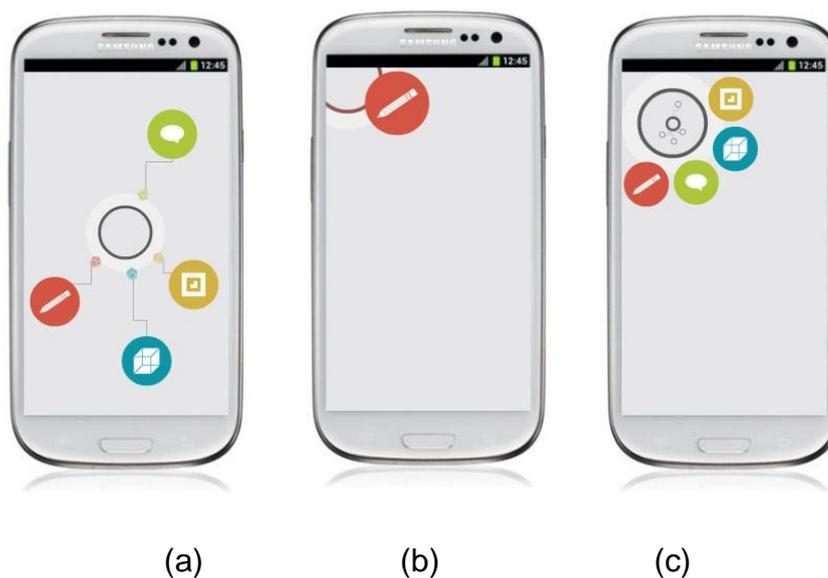


Figura 21 – Formas do menu principal: (a) Exposto; (b) Contraído; (c) Aberto.

Fonte: Acervo pessoal, 2012

5.4 Projeto de Interface

Para se criar um bom projeto de interface para dispositivos móveis é necessário entender os seus potenciais usuários e o que eles buscam (Hoover & Berkman, 2012). Com o constante surgimento de novos aplicativos, os usuários buscam em sua maioria basicamente três características principais:

- Facilidade;
- Agilidade;
- Intuitividade.

Também é necessário entender as limitações dos usuários e da tecnologia utilizada, como os tamanhos das telas disponíveis, os protocolos de comunicação entre dispositivos, etc. É importante pensar que o usuário deve ser capaz de enxergar o conteúdo apresentado sem esforço e que ao utilizar os dedos para interação, parte da tela será ocluída.

5.4.1 Ícones

Os ícones de um aplicativo apresentam forte importância para sua interface, pois fazem parte da sua identidade visual. Se bem desenhado, um ícone pode representar o produto e tornar-se uma “porta de entrada”. Os ícones também apresentam as funções do *app* de maneira visual. Eles ajudam o usuário a retomar uma determinada funcionalidade do sistema (Ware, 2004).

A Figura 22 contém os ícones utilizados para o *app* desenvolvido como prova de conceito deste trabalho. Foram utilizados os guias apresentados por (Ware, 2004) e (Google Inc., 2012) como principais referências em sua criação.

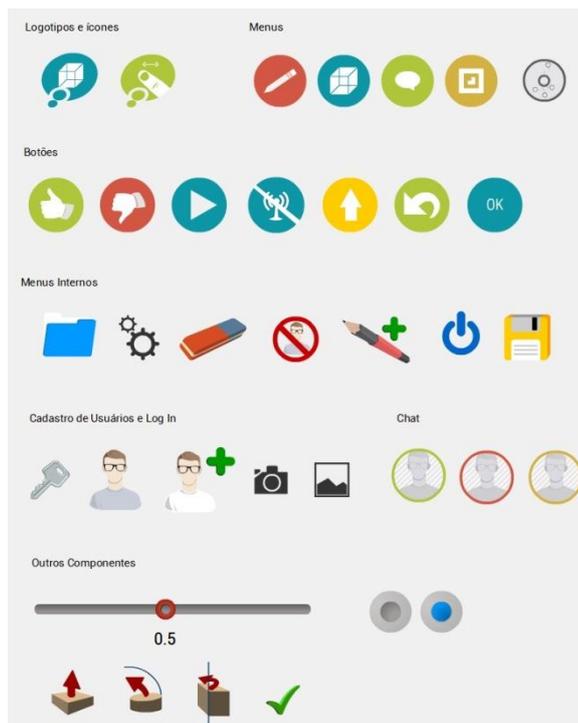


Figura 22 – Ícones desenvolvidos para o aplicativo Imagine 3D

Fonte: Acervo pessoal, 2012

No aplicativo desenvolvido predominam as cores primárias, apresentando desenhos simples e minimalistas, a fim de facilitar a assimilação do mesmo à sua função e seguir as atuais tendências de interfaces.

5.4.2 Padrões de Interação

Padrões de Interação definem alguns controles e elementos para o projeto de interfaces para dispositivos móveis. Bons padrões garantem o fácil uso do aplicativo, de forma intuitiva e simples. Para isso, são necessários padrões para o controle do *app*, ou seja, sua navegação e para os elementos utilizados.

5.4.2.1 Controle

O controle do aplicativo indica a maneira como ele será manipulado e pode ocorrer de duas maneiras principais:

- Manipulação Indireta – feita por meio de componentes físicos como joystick direcional, botões alfa-numéricos, teclado mapeado, etc.

- Manipulação Direta – feita por meio de gestos, toque ou muititoque.

Para o aplicativo em questão, adotaram-se os métodos de manipulação direta, intercalando as funções gestuais e de toque. Basicamente, são usados os toques para simular eventos de clique, gestos com um dedo para inserção de conteúdo e edição dos elementos de um esboço e gestos com dois dedos para transição entre as telas. A criação e o uso de gestos serão explorados no tópico 5.5 deste trabalho.

5.4.2.2 Elementos

São todos os elementos de interação presente nas telas, como botões, listas, *sliders*, *comboboxes*, indicadores de texto e campos de entrada de texto, por exemplo. A Figura 23 apresenta as medidas mínimas adotadas pelo padrão *Android* (Google Inc., 2012).

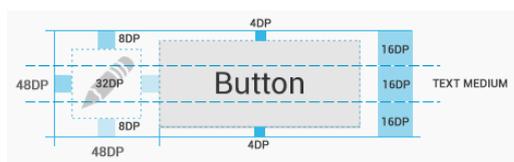


Figura 23 – Métrica de um botão

Fonte: (Google Inc., 2012)

Em interfaces criadas para dispositivos móveis, a métrica dos elementos é muito importante. Tais métricas foram feitas pensando na área necessária para reconhecimento do toque, que exerça um comando específico sem ambiguidade e que permaneça dentro da acuidade visual do usuário.

5.4.3 Flexibilidade

Com a existência de muitos dispositivos diferentes, percebe-se a grande diferença entre eles. Apesar de tentarem seguir padrões de qualidade similares, cada fabricante tem autonomia para determinar o aspecto de seus aparelhos.

Características como tamanho da tela, em polegadas, e sua resolução, em pontos por polegada (DPs) devem ser levadas em consideração na criação do aplicativo.

Uma maneira de garantir que o *app* seja compatível com o maior número de dispositivos é criar uma interface dinâmica e fluida, isto é, uma interface na qual seus

elementos se ajustem em tamanho para que preencham a mesma proporção em telas de diferentes tamanhos. Este procedimento é conhecido como “Design Fluido” e é adotado em muitos aplicativos e páginas web.

5.5 Criação do Sistema Bidimensional

No Imagine 3D, a primeira etapa de criação de um ambiente virtual é o seu rascunho bidimensional. Assim como um rascunho feito com lápis e papel, o usuário deve ser capaz de esboçar formas que se tornarão elementos 3D.

A primeira versão do Imagine 3D, apresentada nas publicações citadas neste trabalho, possui uma biblioteca predefinida de elementos e sua inserção é feita através de métodos de arrastar e soltar, como pode ser visto na Figura 24.

Ao abrir uma biblioteca de figuras pré-determinadas, o usuário é coagido a pensar em um esboço que contenha apenas estas formas, privando-o de usar sua criatividade e desenhar um sistema um pouco mais complexo. Devido a isso, a segunda versão do aplicativo foi pensada de forma a dar total liberdade ao usuário para criar os elementos que desejar.



Figura 24 – Versões do Imagine 3D

Fonte: Acervo pessoal, 2012

A fim de manter o procedimento o mais intuitivo possível, optou-se por criar um ambiente de desenho comum, como os editores conhecidos, de tal forma que, usando toques com um dedo, o usuário cria suas formas como se usasse um lápis sobre o papel.

O processo de desenho e reconhecimento de formas é composto dos seguintes passos:

1. Desenho da forma (entrada do usuário por meio de gestos com um dedo);
2. Simplificação da forma obtida;
3. Reconhecimento da forma simplificada;
4. Substituição ou não da forma original pela simplificada.

Os passos serão descritos e detalhados nos seguintes tópicos ara maior compreensão.

5.5.1 Desenho de formas

Quando o usuário começa seu desenho, o sistema começa a capturar os pontos do caminho esboçado. Ao finalizar o desenho, o sistema apresenta todos os pontos conectados em sequencia, mostrando ao usuário a forma desenhada.

Optou-se por utilizar a linguagem Actionscript 3.0, na plataforma Flash Professional, devido à maior compatibilidade com os navegadores, sistemas operacionais e dispositivos atuais. Por tratar-se de uma linguagem Orientada a Objetos, sua portabilidade futura a outras linguagens não é comprometida e pode ser realizada sem maiores complicações.

Ao adotar o Actionscript e a tecnologia Flash como base para o aplicativo, assumem-se os métodos de programação baseados em eventos. Desta forma, optou-se pela captura dos pontos usando o evento ENTER_FRAME, o que significa que, em um sistema com uma limitada taxa de quadros por segundo, a cada quadro percorrido é guardado um ponto do esboço.

Foi adotada uma taxa de 24 quadros por segundo. Com isso, observa-se que um desenho feito rapidamente, terá menos pontos salvos que um feito lentamente. Para que todos os desenhos tenham a mesma quantidade de pontos e facilitem o reconhecimento, criou-se um algoritmo de simplificação dos pontos.

5.5.2 Simplificação da forma obtida

Seguindo o processo, fez-se necessária a simplificação dos pontos obtidos na primeira etapa. Para isso criou-se um algoritmo baseado no trabalho de Arnaud Icard (Icard, 2010).

Os experimentos foram realizados de maneira modular. Para cada etapa dos mesmos, criaram-se classes genéricas para reutilização posterior. A conexão entre os módulos foi feita na etapa final, precedendo a portabilidade aos sistemas operacionais móveis.

O processo de simplificação de formas visa homogeneizar os caminhos inseridos, para que tenham a mesma quantidade de pontos. Após diferentes testes, encontrou-se um número satisfatório de pontos para o reconhecimento dos gestos. Com uma taxa de 24 quadros por segundo, optou-se por criar caminhos de 64 pontos. Estes valores foram obtidos experimentalmente.

O algoritmo baseia-se na simplificação por intersecção de círculos de raio determinado. Assim, percorre-se o vetor, encontrando novos pontos gerados pela intersecção dos círculos. Para reproduzir o algoritmo podem ser realizados os passos apresentados na Figura 25.

- 1 Cálculo do tamanho do caminho (somatório das distâncias entre os pontos sequencialmente)

```
para i=0 até tamanho(caminho)-1
    tamanho = tamanho + ( pontoi+1 - pontoi )
```

- 2 Determinar o raio dos círculos de intersecção (tamanho do caminho sobre o número de pontos desejados)

```
raio = tamanho/numPts
```

- 3 Percorrer o caminho encontrando os novos pontos (o centro do primeiro círculo é o primeiro ponto. Os próximos centros são os pontos de intersecção)

```
centro = caminho[0];
para i=0 até numPts
    circulo = desenhaCirculo(centro, raio);
    centro = intersecção(caminho, circulo);
```

Figura 25 – Passos para simplificação de padrões

Fonte: Acervo pessoal, 2012

O algoritmo é exemplificado na Figura 26. Observa-se que o caminho gerado para a forma “a cursiva minúscula” possui pontos espaçados de forma irregular devido à diferente velocidade ao desenhar cada trecho do caminho. Ao determinar o raio dos círculos de intersecção, encontra-se um novo caminho com uma quantidade definida de pontos igualmente espaçados.

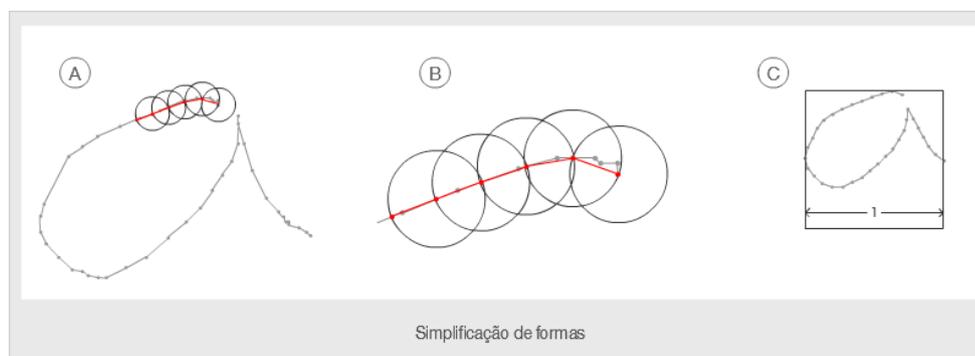


Figura 26 – Simplificação de formas desenhadas

Fonte: Acervo pessoal, 2012

O resultado obtido pelo algoritmo é uma figura com número de pontos determinado pelo usuário. Esta imagem serve de base comparativa para o padrão, portanto, suas unidades de medida são definidas de forma a representarem o objeto em qualquer escala adotada. Para isso, utilizam-se unidades percentuais em vez de unidades métricas comuns como *pixels* ou centímetros, por exemplo. Este procedimento garante que independente do tamanho da figura de entrada, a comparação será realizada com uma mesma escala.

5.5.3 Reconhecimento da forma

Para o reconhecimento das formas adquiridas e simplificadas, optou-se por implementar o algoritmo conhecido como “*\$1 Recognizer*”, apresentado por Wobrock, Wilson e Li (Wobrock, Wilson, & Li, 2007).

O algoritmo apresentou resposta rápida e coerente nos testes realizados por seu autor. Comprovou-se que é um algoritmo para reconhecimento de gestos e formas com uma pequena base de dados.

O método foi escolhido por apresentar codificação clara e não requerer um processamento robusto. O tempo de resposta também é bastante aceitável e pertinente aos propósitos.

Para comprovar a eficiência do algoritmo, criou-se um aplicativo web para captura de desenhos feitos com um mouse, ou outro dispositivo similar, como mesas digitalizadoras ou área de toque de um notebook comum.

Pode-se perceber na Figura 27, na área branca, a forma desenhada pelo usuário. Na área à direita, apresentam-se os resultados obtidos. Percebe-se que o usuário gastou 0,972 segundos para gerar um caminho de 107 pontos. Este foi simplificado com o algoritmo descrito anteriormente e comparado com o banco de dados pré-cadastrado. Observa-se a melhor comparação na área inferior direita.

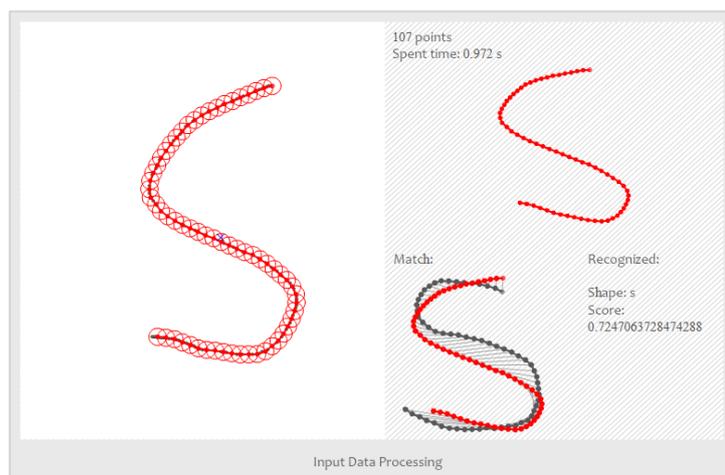


Figura 27 – Algoritmo \$1 Recognizer, prova de conceito.

Fonte: Acervo pessoal, 2012

O algoritmo “\$1 Recognizer” é composto de quatro passos principais:

1. Homogeneização do padrão de entrada;
2. Rotação do padrão para o ângulo indicativo de 0°;
3. Escalar o caminho para uma medida padrão unitário e trasladar o padrão para sua origem;
4. Comparar o padrão obtido com os padrões registrados em um dicionário de formas.

Criou-se assim, uma biblioteca de reconhecimento de formas que pode ser utilizada em qualquer aplicação, definindo seu dicionário de formas.

O dicionário utilizado para a prova de conceito e para o Imagine 3D foi criado utilizando padrões XML para que sejam de fácil construção e acesso tanto em aplicativos web como em aplicativos para *mobile*. Foram acrescentadas as formas mais comuns aos desenhos de modelos físicos: retângulo, quadrado, círculo, triângulo e triângulo retângulo. Também foram adicionadas formas um pouco mais complexas, como: espiral, letra A cursiva minúscula e letra S cursiva minúscula, a fim de verificar a eficácia do método.

A Figura 28 apresenta, como exemplo de forma adicionada ao dicionário, o padrão “s cursiva minúscula”.

```

<PATH name="s" angle="1.9466272095155996">
  <pt x="-79.09145764265088" y="1.1368683772161603e-13"/>
  <pt x="-80.02148826768683" y="8.534786565124875"/>
  <pt x="-79.04734180569909" y="17.059743975609706"/>
  ...
  <pt x="78.97390192590763" y="41.640723942248826"/>
  <pt x="77.33712546427068" y="49.96066735890662"/>
</PATH>

```

Figura 28 – Trecho de código: padrão “S cursiva minúscula”

Fonte: Acervo pessoal, 2012

Observa-se que foram salvos o nome do padrão, o ângulo de rotação ótimo obtido pelo algoritmo e as coordenadas espaciais dos pontos que compõem a forma.

5.5.4 Método alternativo para reconhecimento

Um método alternativo de reconhecimento de padrões foi desenvolvido com a finalidade de comparar os resultados e estabelecer entre eles o mais adequado para o exemplo em questão.

Para criar um novo método de reconhecimento, primeiramente pensou-se nos principais gestos conhecidos e utilizados como padrão nos dispositivos móveis. Estabeleceu-se uma relação entre os gestos e uma codificação gráfica feita através de setas. A Figura 29 apresenta a codificação estabelecida.

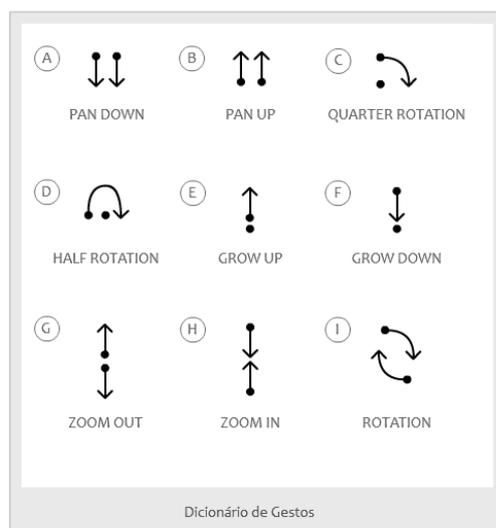


Figura 29 – Codificação dos principais gestos

Fonte: Acervo pessoal, 2012

Como sistema de códigos gráficos, foram criadas flechas que indicam um movimento de dedo, apresentando um ponto na posição onde o movimento começa e uma flecha onde ele termina. Para um simples toque, utiliza-se um ponto.

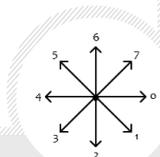
Pensando em um gesto como um conjunto formado por pequenos movimentos, podem ser criados códigos para movimentos complexos.

O primeiro passo do procedimento é a aquisição dos dados do movimento, que é feita da mesma maneira que o método anterior, isto é, uma simples aquisição de coordenadas.

Em seguida, são atribuídos sinais gráficos ao vetor de pontos, tomando os pontos dois a dois e assignando ao trecho entre eles uma forma gráfica correspondente, denominada “sinal”. O dicionário de possíveis sinais é apresentado na Figura 30. Os sinais foram definidos de acordo com o ângulo de inclinação da reta obtida por dois pontos do trajeto. A cada sinal é atribuído um número, definido pela sequência dos ângulos com intervalo de 45° , no sentido horário a partir do eixo horizontal.

Gestos então podem ser interpretados como uma sequência de sinais pré-estabelecidos. Ao se realizar um gesto de meia rotação horária para cima, por exemplo, realiza-se a sequência de sinais: cima, diagonal superior direita, direita, diagonal inferior direita, baixo. O que equivale ao vetor de sinais [6,7,0,1,2]. Contudo, se o gesto for realizado rapidamente, o vetor [7,0,1] também se torna compatível. Se o gesto for ainda mais rápido, o vetor [7,1] é mais uma variação do mesmo gesto.

Sinal	Vetor	Sinal	Vetor	Sinal	Vetor
0. 	[]	10. 	[2,3,4]	20. 	[4,5,6,7,0]
1. 	[6]	11. 	[4,5,6]	21. 	[6,5,4,3,2]
2. 	[2]	12. 	[6,7,0]	22. 	[4,3,2,1,0]
3. 	[0]	13. 	[4,3,2]	23. 	[2,1,0,7,6]
4. 	[4]	14. 	[2,1,0]	24. 	[0,7,6,5,4]
5. 	[7]	15. 	[0,7,6]		
6. 	[5]	16. 	[6,5,4]		
7. 	[1]	17. 	[6,7,0,1,2]		
8. 	[3]	18. 	[0,1,2,3,4]		
9. 	[0,1,2]	19. 	[2,3,4,5,6]		



Dicionário de Sinais

Figura 30 – Dicionário de sinais

Fonte: Acervo pessoal, 2012

Com o intento de reduzir o número de vetores equivalentes por gesto, cria-se um procedimento de simplificação do vetor de sinais, apresentado na Figura 31.

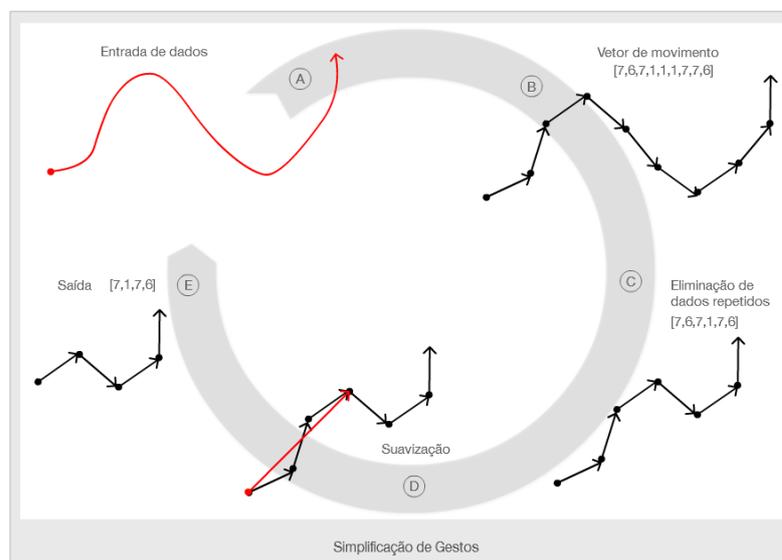


Figura 31 – Processo de codificação de gestos.

Fonte: Acervo pessoal, 2012

As duas primeiras etapas do processo são responsáveis pela aquisição dos dados e representação gráfica. Os demais passos se relacionam ao simplificar do vetor de sinais.

Primeiramente, eliminam os dados repetidos sequencialmente. Com isso, diz-se que a direção e o sentido do movimento são mais importantes que sua magnitude.

Em seguida, acontece o processo de suavização do vetor resultante. Neste processo, buscam-se trechos do vetor que podem ser trocados por um equivalente mais simples. Um movimento composto dos sinais: [cima, diagonal superior direita, cima] pode ser simplificado por [diagonal superior cima], tendo em vista que o ponto de partida e o de chegada são equivalentes nos dois vetores. Esse procedimento é feito apenas para trechos pequenos, para manter a integridade do movimento.

O vetor de saída então é apresentado e comparado com uma base de dados para verificar sua existência.

Este procedimento apresenta, também, bons resultados no reconhecimento. Entretanto, não é possível com ele determinar parâmetros como escala e rotação, que são de suma importância nos métodos de desenho. Por este motivo, o método

não foi escolhido para a etapa de desenho de formas bidimensionais, mas pode ser utilizado para reconhecimento de gestos de controle do aplicativo.

5.5.5 Substituição da Forma Original

Com o intuito de simplificar ainda mais os padrões conhecidos, para um melhor aproveitamento dos métodos de renderização em ambientes virtuais, o usuário tem a opção de substituir o seu padrão de entrada por um padrão ainda mais compacto. Caso ele desenhe um retângulo, por exemplo, e o sistema o reconhece como tal, ele pode substituir seus 64 pontos por apenas quatro que representem os cantos de uma estrutura retangular.

A interação por gestos e toques em dispositivos com telas pequenas pode gerar oclusão de parte do caminho desenhado. Em determinadas situações, o dedo do usuário obstrui a visão completa da tela, gerando desconforto e confusão ao realizar um movimento. A Figura 32 apresenta uma tela de substituição de forma, que dá liberdade ao usuário para escolher entre duas formas semelhantes, de proporções diferentes.

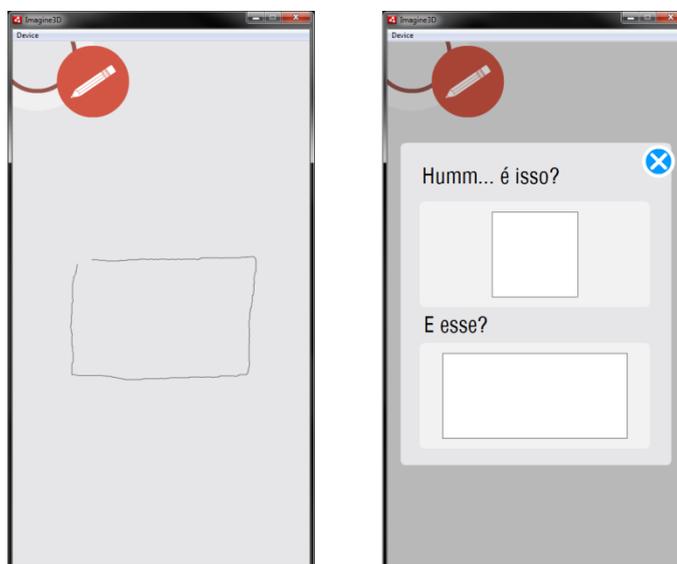


Figura 32 – Telas de substituição de formas.

Fonte: Acervo pessoal, 2012

Nota-se que é utilizado um vocabulário informal, para garantir ainda mais a proximidade com o universo do usuário. Os textos exibidos são aleatórios, mantendo a coerência do discurso.

5.6 Sistema de Realidade Virtual

Quando o usuário finaliza inserção de uma forma, é preciso transformá-la em um modelo tridimensional para que ela seja inserida no ambiente virtual. Primeiramente, avaliou-se, a hipótese de utilização de sequencias de gestos feitos com dois dedos para tal transformação. Entretanto, conclui-se que haveria aumento da complexidade do sistema aumenta perda da intuição, por parte do usuário.

Nos sistemas operacionais para dispositivos móveis já é uma convenção o gesto de pressionar por um intervalo de tempo sobre um objeto para editá-lo (Google Inc., 2012). Assim, para manter o padrão, este gesto foi adotado para permitir ao usuário remover a forma criada ou abrir uma tela de edição do esboço 2D.

A Figura 33 exemplifica o começo da edição da forma básica. Observa-se que ao abrir a tela de edição, o primeiro método de transformação mostrado é uma extrusão linear.

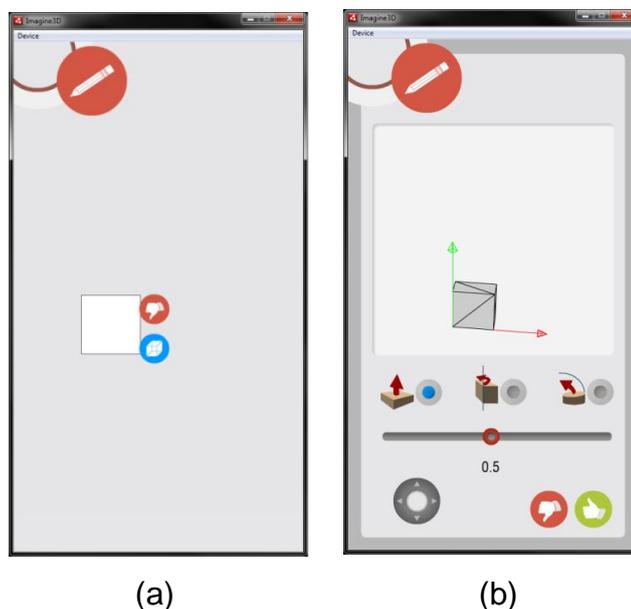


Figura 33 – Edição de formas bidimensionais

Fonte: Acervo pessoal, 2012

Para a transição entre os esboços e os elementos 3D, foram criados três diferentes métodos:

1. Extrusão linear;

2. Rotação de padrão;
3. Extrusão por caminho definido.

Os três métodos apresentam como característica comum a necessidade de um *profile*, isto é, uma forma básica a ser repetida ao longo de um *path*, ou seja um caminho pré-definido.

5.6.1 Extrusão Linear

O método de Extrusão Linear cria um objeto tridimensional a partir da repetição de padrões bidimensionais. Para o *app* Imagine 3D, o padrão utilizado é a forma simplificada, sendo um vetor de pontos.

O caminho neste primeiro caso é uma linha reta, cujo tamanho pode ser alterado através de um componente *slider* que varia de zero ao maior valor entre a largura e a altura do esboço. Essa limitação foi criada especificamente para este aplicativo, a fim de limitar o espaço do ambiente virtual.

O método de extrusão linear pode ser visto na Figura 34 (a). Um ponto importante a ser considerado é a quantidade de pontos do *profile*. Se esta for muito grande, os métodos de renderização do ambiente virtual podem se tornar muito lentos quando processados em um dispositivo móvel.

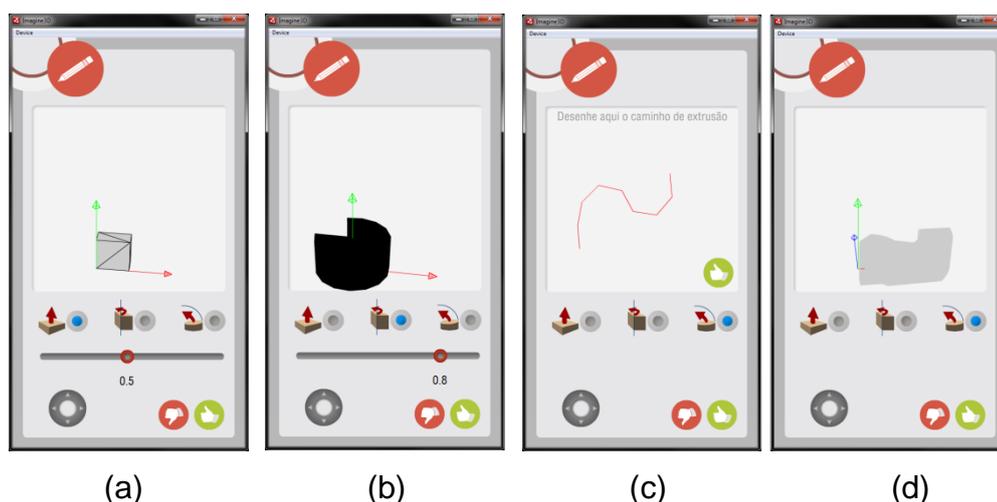


Figura 34 – Métodos de extrusão: (a) linear; (b) rotação; (c) e (d) caminho

Fonte: Acervo pessoal, 2012

Caso o usuário tenha substituído sua forma desenhada por uma predefinida, o método opta por não criar uma extrusão, mas por utilizar um elemento conhecido

na biblioteca de RV utilizada. Um quadrado, por exemplo, pode ser transformado em um cubo em vez de ser extrudido.

5.6.2 Extrusão por rotação - Sweep

A extrusão feita pela rotação de um *profile* segue o mesmo princípio da extrusão linear, diferenciando-se na criação do *path* usado para extrusão. Neste caso, o caminho percorrido é representado por um círculo.

Toma-se a forma de entrada e define-se o seu centro de rotação como o canto superior esquerdo do padrão. Em seguida, rotacional-se o *profile* em um ângulo no intervalo $[0^\circ \ 360^\circ]$. Este ângulo pode ser definido pelo usuário através de um componente *slider* que varia de 0 a 1, correspondendo ao mínimo e ao máximo do intervalo angular.

O resultado de uma rotação de padrão pode ser vista na Figura 34 (b).

5.6.3 Extrusão por caminho definido

O último método de criação de um elemento tridimensional é a extrusão por um caminho definido. O método também é semelhante à extrusão linear, mas o *path* é definido pelo usuário.

A Figura 34 (c) mostra a tela de inserção do *path*. Observa-se que a curva é contínua, mas não suavizada. Isto é feito propositadamente, a fim de tornar ótimo o método de renderização. O caminho desenhado é substituído por um caminho simplificado com dez pontos. Este valor pode ser alterado no painel de configurações do usuário.

Por fim, o elemento 3D é criado percorrendo o *path* gerado. Um exemplo do processo é visto na Figura 34 (d).

5.7 Propriedades Físicas do Ambiente Virtual

Uma vez que todo o sistema esteja esboçado e transformado, o usuário pode alternar entre as telas de desenho e visualização, para acompanhar a ação das forças físicas sobre o sistema.

Ao ser acessada, a tela de visualização apresenta os elementos tridimensionais dispostos de acordo com o esboçado. Para acrescentar uma propriedade física, como peso, velocidade ou aceleração, o usuário utiliza o mesmo sistema de pressionar o elemento por um curto tempo.

Para visualizar a interação física do elemento com o sistema, basta pressionar sobre o botão de reproduzir, localizado no canto inferior direito da tela. Esse procedimento mostra como se comportam os elementos em uma simulação da realidade.

A Figura 35 exemplifica um sistema esboçado e transformado em tridimensional, visualizado com interação das forças físicas.

É importante ressaltar que ao transformar um esboço em 3D, as coordenadas de cada objeto podem precisar de leves alterações. Isso acontece porque nos esboços são levados em consideração apenas os aspectos de altura e largura de uma das faces do elemento, não considerando a profundidade dos mesmos.

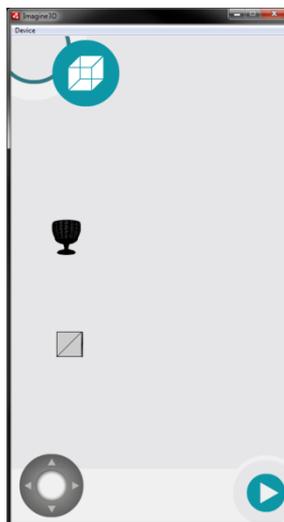


Figura 35 – Tela de visualização do ambiente virtual

Fonte: Acervo pessoal, 2012

5.8 Sistema de Realidade Aumentada

Como forma alternativa de visualização do ambiente construído, o usuário pode alternar para a tela de visualização em Realidade Aumentada.

O princípio de funcionamento é bastante simples. Ao terminar de esboçar seu modelo, transformá-lo em um ambiente virtual e definir suas propriedades físicas, o usuário deve alternar para a visualização em RA.

Nesta tela, a câmera do usuário é acionada e inicia-se o processo de procura por um marcador pré-definido nas configurações do aplicativo. Este marcador pode ser um marcado simples, composto de um desenho simples e assimétrico emoldurado por um quadro preto, ou mesmo uma imagem complexa, como uma foto.

Devido ao grande custo de processamento, este método ainda não foi completamente portado para os dispositivos móveis quando programado com a linguagem Actionscript 3.0 (AS3).

A tradicional biblioteca *FLARToolkit*, utilizada para composições de RA em aplicações web pode ser utilizada, entretanto, percebe-se uma significativa queda na taxa de quadros por segundo. Observou-se ser necessária uma taxa de 5 a 10 quadros por segundo para que o sistema fosse executado nos aparelhos de teste.

Outras bibliotecas vêm sendo portadas para o AS3 visando amenizar esta dificuldade, como as bibliotecas BuildAR, ColAR, do HitLab NZ⁵.

Para a composição do ambiente de RA do Imagine 3D, foi utilizada a biblioteca *FLARToolkit*, devido à sua simplicidade de código, estabilidade (apesar de obter um resultado inferior) e fácil suporte ao desenvolvedor.

Vale ressaltar que o aplicativo foi criado de maneira modular, o que garante a facilidade de alteração do módulo de visualização e interação sem grandes alterações no corpo do aplicativo.

5.9 Comunicação entre Dispositivos

Com o intuito de criar um ambiente colaborativo, pensou-se em um módulo de comunicação entre dispositivos conectados a uma rede.

Baseado em um cenário de utilização do sistema no qual uma turma, em sala de aula, utiliza o Imagine 3D para criar os modelos estudados em aula, pensou-se

⁵ <http://www.hitlabnz.org/index.php/products>

na comunicação entre os dispositivos de cada aluno e do professor, para que cada aluno criasse o seu modelo e permitisse aos demais a sua visualização e alteração. Esta se torna uma prática na qual o professor pode avaliar o desempenho dos alunos e suas dúvidas frequentes.

Pode-se ver na Figura 36 em (a) a comunicação entre dois usuários e em (b) o compartilhamento de modelos.

O compartilhamento de modelos é feito na tela de desenho. Quando o aplicativo é iniciado, o usuário tem a opção de entrar com um usuário cadastrado, cadastrar um novo ou mesmo entrar sem se registrar. Para as duas primeiras opções, a opção “Compartilhar” é liberada no menu físico da tela “Desenho”.

Ao escolher compartilhar, o usuário pode liberar o seu esboço para os demais usuários da rede. Com o mesmo procedimento, pode-se desfazer o compartilhamento.

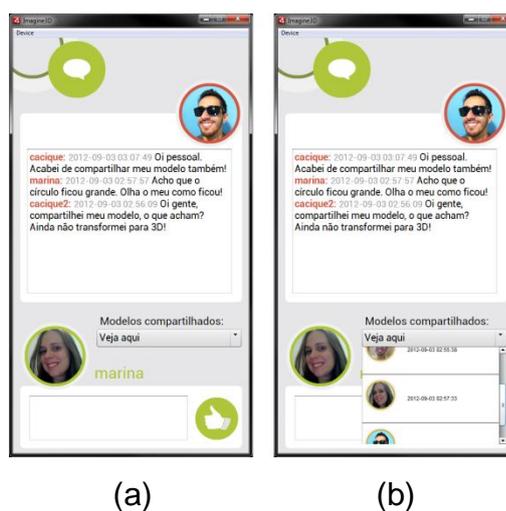


Figura 36 – (a) Comunicação entre dispositivos e (b) compartilhamento de modelos.

Fonte: Acervo pessoal, 2012

Na tela de comunicação, um elemento *comboBox* é inserido para que os modelos compartilhados possam ser abertos e editados. Um modelo editado pode ser salvo em uma diferente versão, para que não seja corrompido.

O *comboBox* apresenta a imagem do usuário criador do modelo e sua data de criação, facilitando a busca pelo modelo desejado.

A tela de comunicação apresenta também um sistema padrão de bate-papo, no qual mensagens são enviadas para todos os usuários. Optou-se por desabilitar a opção de mensagem particular pensando no ambiente escolar, para que o professor tenha o conhecimento do que é dito entre os alunos. Por um lado, a comunicação

entre alunos pode ser benéfica, para o compartilhamento de bons resultados. Entretanto, este fato tende a aumentar a dispersão da turma para assuntos paralelos.

Uma vez que os dispositivos são reconhecidos pelo seu IP, pode ser implementado facilmente um método de bloqueio de mensagens de um usuário caso ele tenha uma conduta que não faz parte. Esse método pode estar visível somente ao professor (administrador do sistema).

Foram levantadas duas opções para a comunicação entre os dispositivos:

- Rede ponto a ponto;
- Rede aberta.

Caso escolhida uma rede ponto-a-ponto, o sistema permitiria apenas a comunicação entre dispositivos em uma rede local com um pequeno alcance (um raio de 30m a 100m em média). Uma rede aberta, em contrapartida, permitiria os usuários compartilhar seus modelos de forma mais ampla e interagir com usuários de outras localidades.

O acesso à Internet pode ser visto como um ponto negativo da escolha por uma rede aberta, pois existem ainda locais no Brasil e em outros países que não contam com essa facilidade disponível.

Ao pesar os aspectos positivos e negativos de cada rede, optou-se pela rede aberta, com o intuito de facilitar a comunicação entre estudantes de diferentes posições sociais e geográficas.

5.10 Instruções ao Usuário

Todo aplicativo desenvolvido para dispositivos móveis deve ter um manual de instruções de uso com fácil acesso. Existem diferentes formas de fazê-lo:

- Por meio de um vídeo expositivo, contendo as principais formas de utilização;
- Com a exibição de um guia interativo, no qual o usuário vai seguindo uma sequência de passos pré-programados, com mensagens de alerta exibindo o erro ou sucesso de uma tarefa realizada;
- Por meio de uma tela específica de ajuda ao usuário (geralmente associada ao botão físico de pesquisa);

Para o Imagine 3D, foi criado um vídeo de explicação das principais funções do *app*. O vídeo pode ser visto no site do aplicativo, no portal de vídeos *Youtube* e também é associado ao botão físico de busca.

A Figura 37 apresenta o vídeo explicativo sendo executado no *player* do portal *Youtube*⁶.

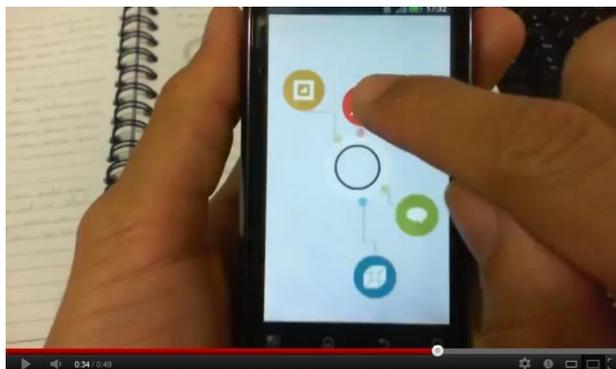


Figura 37 – Vídeo com instruções de uso do Imagine 3D

Fonte: Acervo pessoal, 2012

5.11 Imagine DT – captação de dados comportamentais

Uma maneira encontrada para validar o algoritmo de reconhecimento de formas bidimensionais foi a criação de um segundo aplicativo de desenho. Este *app* tem uma função limitada à captação de dados comportamentais do usuário ao desenhar determinados padrões.

Conhecido como Imagine DT (*Drawing Tester*), o aplicativo não foi publicado em mercados de aplicativos, sendo distribuído apenas entre poucos desenvolvedores, a fim de verificar como os usuários reagem ao desenhar formas geométricas simples em *smartphones*.

Buscou-se criar um aplicativo cuja interação com o usuário fosse simples e intuitiva. O usuário é orientado a realizar quatro desenhos de cada forma geométrica apresentada e que seus desenhos não precisam ficar milimetricamente iguais ao exemplo.

A Figura 38 apresenta a interface gráfica do aplicativo de coleta de dados. Observa-se a coerência entre o projeto de interface deste *app* com o Imagine 3D.

⁶ Vídeo disponível em: <http://youtu.be/4ZY4egZy7V4>



Figura 38 – Interface do aplicativo de coleta de dados: Imagine DT

Fonte: Acervo pessoal, 2012

Quando realizado um desenho, o usuário toca sobre o símbolo verde se julga que o esboço está de acordo, ou sobre o símbolo vermelho caso queira refazer o desenho.

Uma barra lateral mostra quantas formas foram realizadas do padrão em questão. Ao completar as quatro entradas, o usuário deve selecionar o próximo padrão e repetir o procedimento.

Por fim, ao completar os seis padrões de entrada, o usuário preenche os campos de texto com seu nome e email e toca sobre o ícone amarelo para enviar os dados obtidos. Uma mensagem de erro ou sucesso do envio de dados é mostrada ao concluir o processo.

Os dados são enviados a um Banco de Dados (BD) online que abriga os resultados da pesquisa. A base de dados é confidencial e requer identificação apenas do autor para consulta.

A estrutura do BD apresenta apenas duas tabelas, sendo uma para abrigar os dados do usuário (*id_user*, *name* e *email*) e outra para abrigar os dados de cada forma desenhada (*id_shape*, *id_user*, *shape*, *recognized*, *score*, *points*, *time*, *path*).

O diagrama de classes do BD responsável pela coleta de dados pode ser visto na Figura 39.

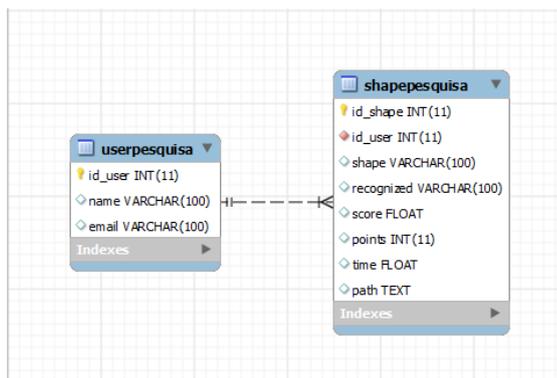


Figura 39 – Diagrama de Classes do Banco de Dados

Fonte: Acervo pessoal, 2012

CAPÍTULO 6

DISCUSSÃO DOS RESULTADOS

6.1 Introdução

Neste capítulo serão apresentados os resultados obtidos na criação e no reconhecimento de gestos personalizados utilizados para esboçar e criar ambientes de realidade virtual e aumentada.

Apresenta-se o aplicativo Imagine DT, elaborado para validação das estratégias de reconhecimento de formas e utilizado para captar dados de diferentes usuários, de forma a analisar o comportamento dos usuários, na geração de formas bidimensionais.

Serão avaliadas, também, formas distintas de comportamentos de diferentes usuários, participantes da captação dos dados e a portabilidade do aplicativo gerado, com vistas à utilização em meio escolar.

6.2 Dados coletados pelo ImagineDT

Uma vez que o sistema de coleta de dados foi projetado, criou-se o sistema de visualização da informação obtida por eles. Uma aplicação web foi criada para apresentar os resultados obtidos.

A primeira tela da aplicação mostra os padrões desenhados por usuário, e pode ser vista na Figura 40. Nesta tela, os vetores de pontos armazenados na tabela ShapePesquisa são apresentados de forma gráfica em cada um dos quadros. Observa-se um ponto preto em cada padrão, que representa o ponto de partida do desenho.



Figura 40 – Visualização dos dados obtidos: Padrões por Usuário

Fonte: Acervo pessoal, 2012

O gráfico de setores, apresentado na Figura 40 mostra o número de erros e acertos, a finalidade da aplicação não se restringe a verificar quantos padrões foram compatíveis com o dicionário, mas verificar o comportamento do usuário ao realizar o desenho. Pode-se abstrair dos dados observados, por exemplo, se as pessoas costumam desenhar seus padrões no sentido horário ou anti-horário. Esta é uma análise relevante para a criação do dicionário final, pois o método “\$1 Recognizer” verifica os padrões exatos, de tal forma que um círculo desenhado no sentido horário é o oposto do círculo desenhado no sentido anti-horário, por exemplo.

Os resultados obtidos com o aplicativo de testes foi apresentado através de gráficos interativos⁷. Foram ao todo 16 usuários cadastrados no banco de dados. Optou-se por utilizar diferentes estereótipos de usuários, considerando idade, sexo e área de atuação. Dentre os usuários cadastrados, tem-se 3 mulheres e 13 homens, todos com idade acima de 18 anos. Do total cadastrado, 5 não pertencem a comunidade de desenvolvedores de software e apenas 3 são desenvolvedores de aplicativos para dispositivos móveis.

Observa-se também, na aplicação, quais foram os padrões com maior quantidade de erros. Ao passar o mouse sobre um padrão desenhado, uma janela do tipo *tooltip*⁸ é mostrada contendo o tempo gasto para realizar o padrão, o número

⁷ <http://www.pedrocacique.com/mestrado/resultados>

⁸ *Tooltip*, ou dica de contexto, é uma moldura *pop up* que se abre ao passar o mouse sobre um elemento.

de pontos utilizados e um sinal verde representando uma boa equiparação ao dicionário ou vermelho para um padrão não encontrado.

A Tabela 3 apresenta os dados obtidos pelos usuários da pesquisa. Por meio da análise desta tabela, conclui-se que 74% dos padrões desenhados foram reconhecidos como similares aos inseridos no dicionário. Apesar de o número de pontos variar no intervalo grande [34 ,147], a média foi de 88 pontos por padrão desenhado, em 2,0165 segundos, isto é, a velocidade de inserção de pontos é cerca de 44 pontos por segundo.

São apresentados os dados de cada usuário na Figura 40, bem como a média de tempo gasto para desenhar os vinte e quatro padrões e a média de pontos gastos para o mesmo procedimento.

Tabela 3 – Dados dos usuários do Imagine DT

Usuário	Média de Tempo (s)	Média de Pontos	Formas reconhecidas (%)
1	0,8350	34	96
2	2,5931	102	75
3	1,8076	74	58
4	1,2810	54	67
5	0,9362	40	92
6	2,0127	81	92
7	4,5098	181	58
8	2,1908	122	67
9	1,7450	74	79
10	3,1319	124	75
11	2,2384	88	71
12	3,7630	147	58
13	0,9936	44	100
14	1,4180	117	58
15	1,1286	46	71
16	1,6799	72	67
Média Geral	2,0165	88	74

Os dados das duas primeiras colunas da Tabela 3 podem ser visualizados nos gráficos⁹ da Figura 41.

⁹ <http://www.pedrocacique.com/mestrado/resultados/resultados2.html>

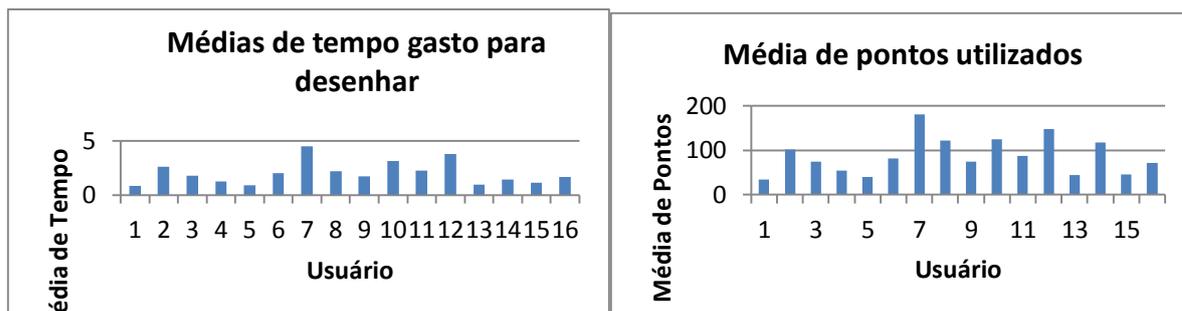


Figura 41 – Gráficos de tempo e pontos por usuário

Fonte: Acervo pessoal, 2012

Observa-se a proporcionalidade dos dados na disposição dos dados. Salva exceção do usuário 14, a forma de ambos os gráficos é similar, o que comprova graficamente o exposto no algoritmo de captura dos pontos: quanto maior o tempo gasto, maior a quantidade de pontos necessários para o reconhecimento.

A Figura 42 apresenta de forma gráfica a porcentagem de erros e acertos de cada forma geométrica.

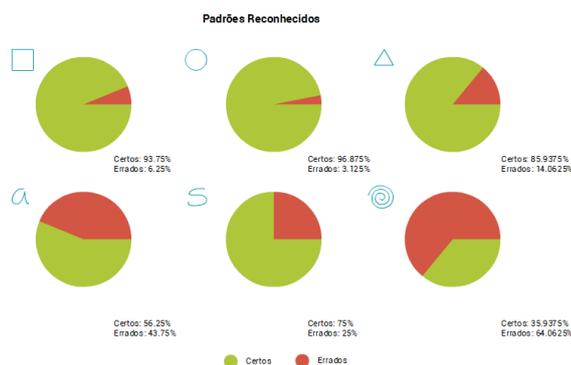


Figura 42 – Gráfico de porcentagem de reconhecimento por padrão

Fonte: Acervo pessoal, 2012

Ao serem analisadas as porcentagens de padrões reconhecidos constatou-se que padrões mais complexos apresentam maior probabilidade de erros. Em especial, analisa-se o padrão “Espiral”, que obteve a maior quantidade de erros. Observando os padrões gráficos obtidos, percebe-se que a maioria dos usuários desenhou espirais com mais de quatro revoluções.

Durante o processo de coleta de dados foi informado ao usuário apenas que ele deveria tentar reproduzir o símbolo apresentado. Apesar de perguntarem

quantas revoluções deveriam ser feitas, a instrução que foi dada não permitia mais detalhes.

Baseado nos dados obtidos, percebe-se que os usuários tendem a verificar o padrão, reconhece-los e desenhá-los como os conhecem, não como uma cópia fiel da imagem apresentada. Esta informação é bastante útil, pois no momento de construção do dicionário deve se pensar o nível de abstração desejado.

O método de reconhecimento utilizado é bastante simples e aplica-se apenas a figuras de baixa complexidade e a um dicionário pequeno, cerca de 10 figuras para bons resultados de reconhecimento (Wobbrock, Wilson, & Li, 2007).

Os padrões adotados para o Aplicativo Imagine 3D foram então escolhido com base nos resultados obtidos pelo Imagine DT. Foram eleitos os padrões “Retângulo”, “Triângulo” e “Círculo”, devido à alta probabilidade de acertos e à relevância para a construção de modelos mecânicos.

Foram analisados, também, métodos de reconhecimento por Redes Neurais Artificiais e Algoritmos Genéticos. Observaram-se métodos simples de AG, utilizando matrizes de pixels para cada padrão obtido. A princípio foram estudados apenas caracteres alfanuméricos. Uma rede perceptron simples também foi sugerida, mas ambos os algoritmos mostraram-se lentos para dispositivos móveis com tempo de resposta cerca de três vezes maior que o tempo de inserção de dados para análise.

CAPÍTULO 7

CONCLUSÕES E TRABALHOS FUTUROS

7.1 Introdução

Neste capítulo, destacam-se os aspectos da pesquisa e conclusões do trabalho apresentado nesta dissertação. Além disso, serão apresentadas algumas sugestões para trabalhos futuros e a contribuição científica do presente trabalho.

7.2 Conclusões

A criação de gestos personalizados requer estudos sobre viabilidade de diferentes métodos de reconhecimento de padrões. Métodos que exigem um nível de complexidade computacional maior podem apresentar resultados mais precisos em arquiteturas de computadores pessoais. Contudo, para dispositivos móveis, que apresentam arquitetura mais simplificada, devem-se ponderar os métodos existentes e optar por aquele que se adeque melhor aos recursos de *hardware* disponíveis.

Os resultados obtidos pelo aplicativo Imagine3D foram satisfatórios. Apesar do método de reconhecimento ser bastante simples, o tempo de resposta obtido foi relativamente pequeno, garantindo a fluidez do uso do *app*. Observa-se que o uso de métodos de inteligência artificial, como algoritmos genéticos ou redes neurais artificiais poderiam trazer resultados mais precisos, mas, devido ao limitado processador dos dispositivos atuais, apresentaria um tempo de resposta alto, caracterizando um aplicativo não usual, assim como apresentando anteriormente.

O uso de dispositivos móveis apresenta um crescimento acelerado e não se restringe apenas à área da telefonia ou do entretenimento, mas abrange diferentes ramificações das mesmas. A Educação é um campo que usufrui do avanço tecnológico e, juntamente com ele, busca novas formas de interação entre os usuários. A sala de aula já não segue o mesmo padrão de alguns anos atrás, mas torna-se cada vez mais dinâmica e não se detém a apenas uma localização geográfica.

A mobilidade da educação já não é mais algo iminente, mas tornou-se realidade nas últimas décadas. Ambientes colaborativos são cada vez mais comuns nas salas de aula e conseqüentemente, o número de aplicativos para *m-learning* cresce em mesma escala.

Os métodos de Realidades Virtual e Aumentada se mostram eficientes no ensino de diferentes disciplinas e sua portabilidade aos dispositivos de comunicação móvel tendem a explorar ainda mais os seus recursos com fins educacionais, descentralizando o aprendizado, outrora focado no professor. O processo cognitivo é favorecido quando se utiliza das tecnologias de comunicação móvel, pois o aluno tem a possibilidade de compartilhar seus conhecimentos com todo o grupo de forma rápida e simples.

Os estudos em *mobile-learning* também acompanham as pesquisas em novas formas de interação e Interfaces Naturais. Tecnologias de interação por meio de gestos e toques são alvos de pesquisas atuais e, associados à educação, buscam novas experiências interativas e coerentes com as práticas modernas de ensino.

7.3 Trabalhos Futuros

Como trabalhos futuros, pode-se pensar em três linhas de ação: aprimoramento do sistema proposto, utilização dos princípios de criação e reconhecimento de gestos para outras finalidades que não a geração de ambientes virtuais e por fim, o uso de ambientes virtuais para auxílio em *m-learning* em diferentes áreas do conhecimento. Com base nestes caminhos, são propostos os trabalhos a seguir.

7.3.1 Aprimoramento do sistema

- Portabilidade do sistema aos diferentes Sistemas Operacionais Móveis existentes;
- Estudo de novos métodos de Realidade Aumentada para computação móvel que aproveitem ao máximo os recursos de *hardware* disponíveis. Pode-se, por exemplo, dividir os processos do SO, deixando a placa gráfica do dispositivo dedicada à renderização enquanto os demais processos ocorrem em paralelo;
- Geração de ambientes de RA que não utilizem marcadores e aprimorem a simulação da interação física dos modelos tridimensionais com o ambiente real;
- Portabilidade do sistema a outras linguagens orientadas a objetos, tais como o Objective-C, Java, Python ou Ruby, por exemplo.

7.3.2 Reconhecimento de gestos para outras finalidades

- Utilização de gestos para a criação de animações tradicionais quadro a quadro. A animação tradicional é baseada na sobreposição de imagens que apresentam leves alterações entre si. A sobreposição das mesmas a uma taxa pequena de quadros por segundo garante a ilusão do movimento. O uso de gestos e toques pode facilitar a organização dos quadros e sua sobreposição, bem como leves alterações nos desenhos propostos.
- Criação de diagramas de Engenharia de Software (ESOF) a partir de esboços rápidos. Existe, atualmente, uma série de softwares voltados à criação de diagramas de ESOF que não são intuitivos ao usuário. Uma possível solução é adaptá-los para que a interação durante a criação de um modelo seja baseada na criação de esboços através de métodos gestuais.
- Simulações de sistemas elétricos, mecânicos e hidráulicos por meio da rápida inserção de elementos. Os diferentes CADs existentes ainda trabalham sobre os padrões de software voltados para computadores pessoais. Entretanto, muitos engenheiros e arquitetos utilizam dispositivos móveis pela facilidade de uso no campo de trabalho externo. Pode ser pensada a portabilidade dos sistemas para o desenvolvimento em dispositivos multitoque.

7.3.3 M-learning para outras áreas do conhecimento

- O ensino de técnicas de artes plásticas, como pintura e escultura pode ser beneficiado com o sistema gestual. A técnica pode ser ensinada em dispositivos móveis com maior sensibilidade de toque para que o aluno possa praticar inúmeras vezes sem que tenha a preocupação em repor o seu material de trabalho, como tintas e pincéis, por exemplo;
- A Geometria Analítica é uma disciplina que requer abstração da informação apresentada, o que não é uma tarefa fácil para todos os alunos. Pode-se pensar em um aplicativo que permita ao professor, em um ambiente colaborativo, criar as formas geométricas, estudadas em sala de aula, em um dispositivo e visualizá-las, em conjunto com os alunos, de diferentes perspectivas e escalas;
- As técnicas de desenho técnico para engenheiros são ensinadas em cursos de graduação com métodos rudimentares que já não se aplicam à realidade dos CADs. Uma solução para este problema é a criação de aplicativos que se assemelhem às interfaces dos sistemas mais comuns na área para que o aluno aprenda as técnicas associadas à prática;

7.4 Considerações Finais

Este trabalho proporcionou a criação de um aplicativo para computação móvel que permite ao usuário desenvolver o seu próprio sistema a partir de esboços gerados por meio de interação gestual e toques. Do ponto de vista científico, as principais contribuições deste trabalho são:

- Avaliação das práticas de projeto de aplicativos para dispositivos móveis;
- Análise do comportamento do usuário ao realizar gestos simples e complexos;
- Proposta de algoritmos eficientes de reconhecimento de padrões de entrada em dispositivos de toque;
- Geração de um aplicativo que pode ser utilizado no auxílio de estudantes de Física do ensino médio aos primeiros anos de cursos das áreas de Exatas.

REFERÊNCIAS BIBLIOGRÁFICAS

- DST, Design Simulation Technologies. (2012). *Working Model*. Acesso em 23 de 05 de 2012, disponível em DST Design Simulation Technologies: <http://www.design-simulation.com/WM2D/download.php>
- Alun-Jones, B. (2010). *Towards a Natural User Interface for Visualization*. Londres: Imperial College London.
- ANATEL. (2011). *Relatório Anual 2011*. Agência Nacional de Telecomunicações, Brasília - DF.
- Apple. (2012). *About iOS*. Acesso em 23 de Julho de 2012, disponível em Apple: <http://www.apple.com/iphone/ios/>
- Apple. (2012). *iOS Developer Library*. Acesso em 23 de Julho de 2012, disponível em iOS Developer Library: <http://developer.apple.com/library/ios/navigation/>
- Azuma, R. (Novembro de 2001). Recent Advances in Augmented Reality. *IEEE Computer Graphics and Applications*, 21(6), 34-47.
- Bae, S.-H., Balakrishnan, R., & Singh, K. (2008). ILoveSketch: As-natural-as-possible system for creating 3D curve models. *Proceedings of the 21st annual ACM symposium on User interface software and technology*, (pp. 151-160). New York.
- Bae, S.-H., Balakrishnan, R., & Singh, K. (2009). EverybodyLovesSketch: 3D Sketching for a Broader Audience. *ACM Symposium on User Interface Software and Technology*, (pp. 59-68). Victoria, BC, Canada.
- Belloni, M. L. (2008). *Educação a distância (Coleção Educação Contemporânea)* (5ª ed.). Campinas: Autores Associados.
- Bergig, O., Hagbi, N., El-Sana, J., & Billinghamurst, M. (2009). In-place 3D sketching for authoring and augmenting mechanical systems. *ISMAR '09 Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality* (pp. 87-94). Washington DC: IEEE Computer Society.
- Bergig, O., Hagbi, N., El-Sana, J., & Billinghamurst, M. (2009). In-Place 3D Sketching for Authoring and Augmenting Mechanical Systems. *ISMAR '09 Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality*, pp. 87-94.
- Bhuiyan, M., & Picking, R. (2009). Gesture-controlled user interfaces, what have we done and what's next? *Fifth Collaborative Research Symposium on Security*,

- E-Learning, Internet and Networking (SEIN 2009)*, (pp. 59 - 60). Darmstadt, Germany.
- Billinghamurst, M., Kato, H., & Poupyrev, I. (2001). The Magic Book - Moving Seamlessly between Reality and Virtuality. *Computer Graphics and Applications*, 21(3), 2-4.
- Bragdon, A., Nelson, E., Li, Y., & Hinckley, K. (2011). Experimental Analysis of Touch-Screen gesture Designs in Mobile Environments. *CHI 2011*. Vancouver, BC, Canadá.
- Bragdon, A., Nelson, E., Li, Y., & Hinckley, K. (2011). Experimental Analysis of Touch-Screen Gesture Designs in Mobile Environments. *CHI2011*, (pp. 403-412). Vancouver, BC, Canada.
- Bullinger, H. J., Rößler, F., & Müller-Spahn, A. (1996). Encouraging Creativity - Support of Mental Processes by Virtual Experience. *IDG Conferences & Seminars*.
- Campos, I. M. (2006). *Relatório inicial dos resultados da TIC DOMICILIOS e USUÁRIOS*. Acesso em 17 de Agosto de 2011, disponível em Comitê Gestor da Internet no Brasil: <http://www.cetic.br/tic/2006/relatorio-inicial-2006.ppt>
- Castells, M. (1999). *A Sociedade em Rede* (Vol. 1). São Paulo: Paz e Terra.
- Davis, R. (2007). Magic Paper: Sketch-Understanding Research. *IEEE Computer Society*, pp. 34-41.
- Feldmann, M. G. (2005). *Educação e mídias interativas: formando professores*. São Paulo: EDUC - Editora da PUC-SP.
- Gartner. (2012). *Gartner Says Worldwide Sales of Mobile Phones Declined 2 Percent in First Quarter of 2012*. Acesso em 19 de julho de 2012, disponível em Gartner: <http://www.gartner.com/it/page.jsp?id=2017015>
- Gonzalez, R. C., & Woods, R. (2000). *Processamento de Imagens Digitais*. Edgard Blucher Editó.
- Google Inc. (2012). *Android About*. Acesso em 23 de Julho de 2012, disponível em Android: <http://www.android.com/about/>
- Google Inc. (2012). *Android Design*. Acesso em 23 de Julho de 2012, disponível em Android Design: <http://developer.android.com/design/index.html>
- Hofman, M., Hock, C., & Müller-Spahn, F. (1995). Computer-based cognitive training in Alzheimer's disease patients. . *New York Academy of Science: J .H. Gordon, R. Nitsch, J. Wurtmann*.

- Hooper, S., & Berkman, E. (2012). *Designing Mobile Interfaces* (1ª ed.). (M. Treseler, Ed.) Sebastopol, CA: O'Reilly Media.
- Icard, A. (2010). *Simple gesture Recognition in AS3*. Acesso em 23 de 08 de 2011, disponível em Coder's Journal of Arnaud Icard: <http://blog.sqrtof5.com/?p=173>
- Isaacson, W. (2011). *Steve Jobs, The Biography*. Simon & Schuster.
- Jesus, R. P., Fernandes, A. M., Dazzi, R. L., & Moreira, B. G. (2008). Colibri – Ambiente Virtual de jogos Educativos que Auxilia na Reabilitação Cognitiva de Pessoas Portadoras de Deficiência Mental. *Hífen*, 32(62), 308-313.
- Kirner, C., & Kirner, T. G. (2008). Virtual Reality and Augmented Reality Applied to Simulation Visualization. Em A. El Sheikh, A. T. Al Ajeeli, & E. M. Abu-Taieh, *Simulation and Modeling: Current Technologies and Applications* (Vol. 1, pp. 391-419). Hershey-NY: IGI Publishing.
- Kirner, C., & Zorzal, E. R. (2005). Aplicações Educacionais em Ambientes Colaborativos Realidade Aumentada. *XVI SBIE2005 - Simpósio Brasileiro de Informática na Educação*. Juiz de Fora.
- Kubota, L., Demoliner, V., & Figueiredo, C. d. (2011). Um Perfil do Uso da Educação On-line no Brasil”,. *Radar. Tecnologia, Produção e Comércio Exterior, Diretoria de Estudos e Políticas Setoriais, de Inovação, Regulação e Infraestrutura*, 13, 15-21.
- Lü, H., & Li, Y. (2011). Gesture Avatar: A Technique for Operating Mobile User Interfaces Using Gestures. *CHI 2011*, (pp. 207-216). Vancouver, BC, Canada.
- MEC. (2010). *Resumo Técnico Censo da Educação Superior de 2009*. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira - INEP, Brasília - DF.
- Microsoft. (2012). *Windows User Experience Interaction Guidelines*. Acesso em 05 de Setembro de 2012, disponível em Microsoft Download Center: <http://www.microsoft.com/en-us/download/confirmation.aspx?id=2695>
- Microsoft Research. (2012). *Physics Illustrator for Tablet PC Source Code*. Acesso em 01 de 06 de 2012, disponível em Microsoft Research: <http://research.microsoft.com/en-us/downloads/ae3085-a219-47d6-88fc-a2501f00800d/default.aspx>
- Morris, B. (2007). *The Symbian OS Architecture Sourcebook - Design and evolution of a Mobile Phone OS*. West Sussex, England: John Wiley & Sons Ltd.

- Sa, M. d., & Churchill, E. F. (2012). Mobile Augmented Reality: Exploring Design and Prototyping Techniques. *Mobile HCI'12*. San Francisco, CA, USA.
- Shi-Hong, J., Hong, Y. M., Chin, C. H., Ming, Y. C., & Yao, T. L. (Março de 1998). Facial feature detection using geometrical face model: An efficient approach. *Pattern Recognition*. *Pattern Recognition*, 31(3), 273-282.
- Tanembaum, A. S. (2008). *Sistemas Operacionais Modernos* (3ª ed.). São Paulo: Pearson Education do Brasil.
- Tori, R., Kirner, C., & Siscoutto, R. (2006). Fundamentos e Tecnologia de Realidade Virtual e Aumentada. *VIII Symposium on Virtual Reality*. Belém, PA.
- Tucker, T. G., & Winchester III, W. W. (2009). Mobile Learning for Just-In-Time Applications. *ACM-SE 47 Proceedings of the 47th Annual Southeast Regional Conference*.
- Ware, C. (2004). *Information Visualization: Perception for Design* (2ª ed.). San Francisco, CA: Morgan Kaufmann, Elsevier.
- Wobbrock, J. O., Wilson, A. D., & Li, Y. (2007). Gestures without libraries, toolkits or training: A \$1 Recognizer for User Interface Prototypes. *ACM*, pp. 159-168.