



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CARLOS GUSTAVO RESQUE DOS SANTOS

**Modelo Adaptativo às Transições entre Ambientes Internos
e Externos para Aplicações de Realidade Aumentada Móvel**

Belém
2017

Carlos Gustavo Resque dos Santos

Modelo Adaptativo às Transições entre Ambientes Internos e Externos para Aplicações de Realidade Aumentada Móvel

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação. Instituto de Ciências Exatas e Naturais. Universidade Federal do Pará.

Orientador Prof. Dr. Bianchi Serique Meiguins

Belém

2017

Dados Internacionais de Catalogação - na - Publicação (CIP)
Biblioteca de Pós-Graduação do ICEN/UFPA

Santos, Carlos Gustavo Resque dos
Modelo adaptativo às transições entre ambientes internos
e externos para aplicações de Realidade Aumentada Móvel/
Carlos Gustavo Resque dos Santos; orientador, Bianchi Serique
Meiguins.-2017.

129 f. il. 29 cm

Inclui bibliografias

Tese (Doutorado) – Universidade Federal do Pará, Instituto
de Ciências Exatas e Naturais, Programa de Pós-Graduação em
Ciência da Computação, Belém, 2017.

1. Tecnologia da informação. 2. Realidade aumentada-
Modelos-Adaptação-Avaliação. 3. Sistema de Posicionamento
Global. 4. Detectores-Inovações tecnológicas. 5. Dispositivos
Móveis. I. Meiguins, Bianchi Serique, orient. II.Título.

CDD – 22 ed. 303.4833

Carlos Gustavo Resque dos Santos

Modelo Adaptativo às Transições entre Ambientes Internos e Externos para Aplicações de Realidade Aumentada Móvel

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação. Instituto de Ciências Exatas e Naturais. Universidade Federal do Pará.

Data da defesa: Belém, 9 de junho de 2017

Prof. Dr. Bianchi Serique Meiguins – Orientador
Programa de Pós-Graduação em Ciência da Computação – UFPA

Prof. Dr. Marcelo de Paiva Guimarães – Membro Externo
Universidade Federal de São Paulo (UNIFESP) – São Paulo, SP

Prof. Dr. Schubert Ribeiro de Carvalho – Membro Externo
Instituto Tecnológico Vale (ITV) – Belém, PA

Prof. Dr. Nelson Cruz Sampaio Neto – Membro Interno
Programa de Pós-Graduação em Ciência da Computação – UFPA

Prof. Dr. Jefferson Magalhães de Moraes – Membro Interno
Programa de Pós-Graduação em Ciência da Computação – UFPA

AGRADECIMENTOS

Agradeço ao meu orientador Prof. Dr Bianchi Serique Meiguins por me aceitar como orientando, por me fornecer ensinamentos e orientações que me iluminaram durante o doutorado, que levarei para minha vida pessoal e profissional.

Agradeço à equipe do Laboratório de Visualização da Informação (LABVIS) da UFPa por todas as atividades realizadas em equipe que me ajudaram a concluir este trabalho – em especial ao Tiago Araújo por ser um fiel amigo e dedicar muito tempo da sua vida contribuindo com a realização desta tese. Agradeço ainda em especial aos membros Brunelli Miranda, Alexandre Freitas e Paulo Chagas Jr. por serem grandes amigos além de colegas de trabalho.

Agradeço à douta banca avaliadora constituída pelos professores Marcelo de Paiva Guimarães, Schubert Ribeiro de Carvalho, Nelson Cruz Sampaio Neto e Jefferson Magalhães de Moraes por contribuições, críticas construtivas e questionamentos dados, que me fizeram refletir e aprimorar esta tese de doutorado.

Agradeço aos meus pais Carlos Augusto e Sofia Resque, à minha noiva Aline Barbosa e a meu irmão Roosevelt Resque por seu apoio, carinho, dedicação e compreensão que me impulsionam a cada dia.

Agradeço a todos que pacientemente realizaram os testes presentes neste trabalho e a todos que participaram indiretamente.

Agradeço à CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo financiamento da minha bolsa de doutoramento e à empresa Gol Software pelo espaço gentilmente disponibilizado nos finais de semana para o desenvolvimento de atividades acadêmicas do grupo LABVIS.

*“Come together, yeah
Come together, yeah
Come together, yeah
Come together, yeah
Come together, yeah
...”*

(The Beatles)

RESUMO

A adaptação às mudanças do ambiente físico, principalmente nas transições entre ambientes interiores e exteriores, é um dos desafios atuais da área da Realidade Aumentada Móvel (RAM). Visto que essas transições causam alterações no mundo real, como na iluminação que dificulta o reconhecimento de lugares, ou na existência de obstáculos que dificultam a comunicação com sensores, reduzindo a acurácia da localização do usuário. De maneira geral, as aplicações de RAM ou estão voltadas somente para ambientes exteriores com registro realizado por meio de GPS (*Global Positioning System*) e sensores inerciais, ou somente para ambientes interiores com registro realizado com base no reconhecimento de imagens. Considerando o contexto exposto, esta tese apresenta um modelo adaptativo às transições entre ambientes interiores e exteriores para aplicações de RAM. O modelo possui um enfoque híbrido, elegendo a melhor combinação entre sensores de longo alcance (GPS), sensores de curto alcance (Wi-Fi, Beacons, entre outros) e técnicas de visão computacional, para mitigar (de forma transparente para o usuário) os problemas de localização, orientação e registro. Para validar o modelo, foi desenvolvida uma aplicação de RAM e foram realizadas avaliações na acurácia de classificação dos ambientes, acurácia da localização, velocidade e acurácia do reconhecimento de imagens. Por fim, foram realizadas avaliações de usabilidade com avaliadores experientes em tecnologia da informação através de uma etapa de experimentação e entrevista posterior. Os resultados mostram que a aplicação de fato se adaptou aos ambientes e foi considerada como uma aplicação que auxilia os usuários a descobrirem POIs (*Points of Interest*), apesar de apresentar problemas de precisão no GPS e demora para atualizar as informações para os usuários.

Palavras-chave: Realidade Aumentada Móvel, Modelo Adaptativo, Registro, GPS, Sensores Inerciais, Reconhecimento de Imagem.

ABSTRACT

Adapting to changes in the physical environment, especially in the transitions between indoor and outdoor environments, is one of the current challenges in the area of Mobile Augmented Reality (MAR). Since these transitions cause changes in the real world, such as in lighting that makes it difficult to recognize places, or in the existence of obstacles that hamper communication with sensors, reducing the accuracy of the user's location. In general, the applications of MAR are oriented only to outdoor environments with registration made using GPS (Global Positioning System) and inertial sensors, or oriented only for indoor environments with registration based on image recognition. Hence, this thesis presents an adaptive model to the transitions between indoor and outdoor environments for MAR applications. The model has a hybrid approach, choosing the best combination of long-range sensors (GPS), short-range sensors (Wi-Fi, Beacons, among others) and computer vision techniques to mitigate (transparently to the user) the problems of location, orientation, and registration. A MAR application was developed to validate the model and evaluations were performed on the accuracy of the environment classification, accuracy of the location, speed and accuracy of image recognition. Finally, usability evaluations were performed with specialists through a stage of experimentation and subsequent interview. The results show that the application self-adapt to the environments and was considered as an application that helps the users to discover POIs (Points of Interest), although it presents problems of precision in the GPS and takes much time to update the information for the users.

Key-words: Mobile Augmented Reality, Adaptive Model, Register, GPS, Inertial Sensors, Image Recognition.

LISTA DE ILUSTRAÇÕES

Figura 1. Contínuo Realidade-Virtualidade adaptado de Milgram et al. (1994).	33
Figura 2. Cenário típico de um ambiente de Realidade Aumentada. Adaptado de (Bimber, Raskar, 2005).	34
Figura 3. Tipos de <i>display</i> pela sua posição. Adaptado de Bimber e Raskar (2005).....	35
Figura 4. Arquiteturas identificadas por Butchart (2011). As arquiteturas são: Plataforma (a), Web (b), Independente (c).	36
Figura 5. Funcionamento do algoritmo proposto por Zheng et al. (2013). A esquerda é mostrada a imagem e pontos utilizados como referência e à direita é mostrado o a saída do algoritmo para uma foto.....	40
Figura 6. Duas visões do resultado final do algoritmo PTAM (Klein, Murray, 2009) (cima) aplicado a uma sequência de imagens de um vídeo (baixo).	42
Figura 7. Aplicação proposta por Dong et al. (2013). As figuras acima mostram o objeto plano que está sendo reconhecido e um eixo virtual. As figuras abaixo mostram a aplicação funcionando.....	43
Figura 8. Aplicação proposta por Brata, Liang e Pramono (2015). Os marcadores apontam a posição do ônibus (verde) e parada (vermelho). À esquerda-acima é mostrado um radar.....	45
Figura 9. Aplicação proposta por Carmo et al. (2016) à esquerda. À direita é mostrada com detalhes a técnica de agregação proposta.....	45
Figura 10. Ao invés de calcular a orientação das características da imagem baseado somente na intensidade dos pixels (esquerda), Kurz e Benhimane (2012a) propuseram a utilizam da gravidade para obter a orientação (direita). Abaixo o aplicativo Junaio é mostrado com a proposta aplicada (baixo).....	46
Figura 11. Na interação inserida no mundo misturado as ações ocorrem depois do dispositivo de mão (a). Na interação associada ao dispositivo as ações ocorrem manipulando o dispositivo (b).....	47
Figura 12. Taxonomia proposta para a organização dos gestos encontrados na literatura.	48

Figura 13. Aplicação MarkerPen (Seidinger, Grubert, 2016) que utiliza uma caneta e um marcador fiducial para realizar transformações (translação, escala e rotação) em avatares 3D. ...	49
Figura 14. Aplicação de Choi et al. (2011) reconhecendo o gesto feito com a mão para selecionar um item de um menu (a). À direita da figura é mostrada a mão segmentada da imagem original (c) e o resultado da identificação dos dedos (b).	49
Figura 15. Resumo visual do trabalho de Pollmann et al. (2013) mostrando a aplicação implementada (A), as interações de mira (B), movimentação relativa (c) e o estudo realizado sobre o efeito de desalinhamento que essas interações causam (D).	50
Figura 16. <i>Pipeline</i> resumido de RA com a adição de um módulo de controle adaptativo.....	57
Figura 17. Modelo adaptativo às transições do usuário entre ambientes internos e externos.	58
Figura 18. Em (a) é ilustrado dois níveis de AOIs, em (b) é ilustrado o AOI azul de (a) com seus POIs internos, e em (c) é ilustrada a planta baixa dos dois andares presentes no POI vermelho ilustrado em (b).	62
Figura 19. Diagrama de casos de uso. Em azul os casos de uso adaptativos, e os casos de uso descritos com “...” indicam uma sequência de exemplos de casos de uso.	64
Figura 20. Visão geral dos diagramas de classe.	65
Figura 21. Diagrama de classes da localização do usuário em relação aos <i>Geofences</i>.	67
Figura 22. Diagrama de classes do reconhecimento de imagens associadas à POIs.	68
Figura 23. Diagrama de classes do serviço de localização interna.	69
Figura 24. Visão geral dos diagramas de sequência. * SLA (Sensor de Longo Alcance).....	70
Figura 25. Diagrama de Sequência da lógica de localização do usuário em relação aos <i>Geofences</i> dos POIs e AOIs.	71
Figura 26. Diagrama de sequências da lógica que verifica se há disponibilidade de localização interna no POI atual e inicia o serviço em caso positivo.	72
Figura 27. Diagrama de sequência do serviço de localização interna.	73
Figura 28. Diagrama de sequência do serviço de reconhecimento de imagem.	74
Figura 29. Diagrama de Sequência de Erro no GPS	75
Figura 30. Interface gráfica da aplicação protótipo. (a) mostra a aplicação funcionando em um ambiente externo e seguindo uma rota, (b) mostra a aplicação funcionando em ambiente interno e reconhecendo uma placa associada a um POI.	78

Figura 31. Em (a) é apresentada a lista de POIs que foram encontrados com a busca para palavra “instituto” e em (b) é apresentado o conteúdo do POI “Reitoria” onde é possível ler um texto sobre o POI, ver imagens e traçar uma rota até o POI.	79
Figura 32. Organização hierárquica dos POIs utilizados no protótipo de validação.	80
Figura 33. Capturas de tela do protótipo de validação no setor profissional apontando para o setor básico (a), no setor básico apontando para o prédio do ICEN (b), dentro do ICEN, apontando para a sala FC-2 (c) e reconhecendo a placa da sala (d).	81
Figura 34. Planta dos dois andares do ICEN com as localizações discretas (letras maiúsculas) utilizadas como treinamento e saída dos classificadores da posição interna. Para cada localização foram registradas as potências do sinal dos Aps de Wi-Fi para serem usados no treinamento do classificador.	84
Figura 35. Imagem em escala de cinza e suas características detectadas e representadas por círculos coloridos.	86
Figura 36. Imagem de cena (esquerda), imagem de referência (direita) e as correspondências de suas características (linhas coloridas).	88
Figura 37. <i>Pipeline</i> do registro baseado em visão (seta Imagem) ao lado do registro baseado em sensores (seta Sensores).	90
Figura 38. Gráficos dos testes de tempo e precisão mostrando os valores utilizados para escalar as imagens utilizadas nos testes e os algoritmos que foram comparados.	91
Figura 39. Gráfico do tempo médio para realizar o registro quando se utiliza de forma alternada reconhecimento de imagem e sensores. O valor das abcissas é número de n° de frames que utilizam sensores ao invés de reconhecimento de imagem.	92
Figura 40. Visão geral dos caminhos registrados pela aplicação medida por GPS, durante o trajeto realizado pelos participantes (linhas vermelhas) em comparação com os caminhos registrados manualmente por um condutor do teste (linha azul). O caminho registrado pelo condutor do teste serve de referência.	98
Figura 41. Destaque do trajeto realizado por dois participantes do teste. Nestes é possível observar uma maior variação nas extremidades da localização e uma falha no posicionamento do GPS (b).	99
Figura 42. Visão geral do caminho feito pelos participantes dentro do prédio do ICEN. As linhas vermelhas representam a posição dos participantes obtidas pela localização com sinais de Wi-Fi e as linhas azul foram registradas por um condutor do teste.	99
Figura 43. Destaque do caminho de um participante dentro do prédio do ICEN, mostrando o momento em que o participante sobe a escada (B) e ocorre uma demora em atualizar sua localização, fazendo com que a aplicação dê um salto com a localização do usuário.	100

Figura 44. Registro da localização medida pelo GPS quando o usuário está dentro do ICEN. Observa-se que é inviável utilizar o GPS em ambientes internos como o ICEN.	101
Figura 45. <i>Tag cloud</i> das palavras mais utilizadas pelos participantes durante a entrevista.	101
Figura 46. Destaque de frases relacionadas com a palavra “marcador”.	102
Figura 47. Destaque de frases relacionadas com a palavra “localização”.	103
Figura 48. Destaque de frases relacionadas com a palavra “perceber”.	103
Figura 49. Destaque de frases relacionadas com a palavra “mudança”.	104
Figura 50. Relacionamento da qualificação das frases (eixo esquerdo) em relação às atividades que as frases se referiram (eixo direito).	105
Figura 51. Relacionamento das atividades que as frases dos participantes se referem (eixo esquerdo) em relação aos problemas observados (eixo direito). Essa visualização mostra apenas as frases que comentavam sobre um ou mais problemas.	107
Figura 52. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	108
Figura 53. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	108
Figura 54. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	109
Figura 55. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	110
Figura 56. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	111
Figura 57. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	111
Figura 58. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.	112

LISTA DE TABELAS

Tabela 1. Comparação entre as funcionalidades das aplicações selecionadas. A cor verde com um “✓” significa que a ferramenta possui a funcionalidade, a cor vermelha clara significa que não há nenhuma funcionalidade do grupo inteiro e quando não há cor significa que a ferramenta não possui essa funcionalidade, mas possui outras desse mesmo grupo de funcionalidades.	54
Tabela 2. Métricas obtidas pelo treinamento dos classificadores para obter a localização interna.	85
Tabela 3. Roteiro de perguntas feitas aos participantes do teste.....	96

LISTA DE SIGLAS

AKAZE	<i>Accelerated KAZE</i>
AOI	<i>Area of Interest</i>
BA	<i>Bundle Adjustment</i>
BRIEF	<i>Binary Robust Independent Elementary Features</i>
BRISK	<i>Binary Robust Invariant Scalable Keypoints</i>
DLT	<i>Direct Linear Transform</i>
EKF	<i>Extended Kalman Filter</i>
FAST	<i>Features from Accelerated Segment Test</i>
FED	<i>Fast Explicit Diffusion</i>
GPS	<i>Global Positioning System</i>
GUI	<i>Graphical User Interface</i>
ICEN	<i>Instituto de Ciências Exatas e Naturais</i>
IMU	<i>Inertial Measurement Unit</i>
IOI	<i>Item of Interest</i>
LABVIS	<i>Laboratório de Visualização, Interação e Sistemas inteligentes</i>
LMEDS	<i>Least MEDian of Squares</i>
MAR	<i>Mobile Augmented Reality</i>
ORB	<i>Oriented FAST and rotated BRIEF</i>

POI	<i>Point of Interest</i>
PTAM	<i>Parallel Traking And Mapping</i>
QR Code	<i>Quick Response Code</i>
RA	Realidade Aumentada
RANSAC	<i>RANdom SAmples Consensus</i>
RFID	<i>Radio-Frequency IDentification</i>
RM	Realidade Misturada
RV	Realidade Virtual
SDK	<i>Software Development Kit</i>
SLAM	<i>Simultaneous Localization And Mapping</i>
UFPA	Universidade Federal do Pará
UML	<i>Unified Modeling Language</i>
VA	Virtualidade Aumentada
WIMP	<i>Window, Icon, Menu, Pointing device</i>

SUMÁRIO

1. INTRODUÇÃO	25
1.1 Justificativa	27
1.2 Objetivos.....	28
1.3 Contribuições	28
1.3.1 Publicações Relacionadas com a Tese.....	29
1.3.2 Publicações Submetidas.....	29
1.3.3 Publicações aceitas (em andamento para publicação final).....	29
1.3.4 Publicações Realizadas Durante o Doutorado	30
1.4 Organização do Trabalho	30
2. REALIDADE AUMENTADA	33
2.1 Conceitos e Teoria	33
2.1.1 Sistemas de Realidade Aumentada.....	34
2.1.2 Estilos Arquiteturais em RA.....	36
2.1.3 Realidade Aumentada Móvel	37
2.2 Algoritmos de Registro.....	38
2.2.1 Registro Baseado em Visão	38
2.2.2 Registro Baseado em Sensores	43
2.2.3 Registro Híbrido	45

2.3	Interação em aplicações de RAM.....	47
2.3.1	Interações Inseridas no Mundo Misturado	48
2.3.2	Interações associadas ao dispositivo.....	50
2.4	Aplicações e Frameworks Relacionados.....	51
2.4.1	Awila.....	51
2.4.2	Augment	51
2.4.3	Junaio.....	51
2.4.4	Layar	52
2.4.5	Blippar	52
2.4.6	Wikitude	52
2.4.7	Vuforia.....	52
2.4.8	Comparação	53
3.	MODELO DE ARQUITETURA	57
3.1	Descrição do Modelo	57
3.2	Localização.....	60
3.2.1	Externa.....	60
3.2.2	Transição.....	62
3.2.3	Interna.....	63
3.3	Aspectos de Implementação.....	63
3.3.1	Diagrama de Caso de Uso.....	63
3.3.2	Diagrama de Classes.....	65
3.3.3	Diagramas de Sequência.....	69
4.	PROTÓTIPO DE VALIDAÇÃO	77
4.1	Interface da Aplicação	77
4.2	Organização Hierárquica do Cenário de Teste	79

4.3	Navegação com RA.....	80
4.4	Lógica Adaptativa	82
4.5	Localização Interna	83
4.6	Algoritmos de Registro baseado em Visão	86
4.6.1	Detecção e Descrição de Características	86
4.6.2	Correspondência entre Características.....	88
4.6.3	Transformação Homográfica	88
4.7	Reconhecimento de Imagens	89
4.8	Reconhecimento de Cena Interno-Externo	93
5.	AVALIAÇÃO DA APLICAÇÃO	95
5.1	Organização do Teste	95
5.2	Registro do Percurso do Teste.....	97
5.3	Resultados da Entrevista	101
	CONCLUSÃO	113
	REFERÊNCIAS	117

1. INTRODUÇÃO

A Realidade Aumentada (RA) ao combinar elementos virtuais a cenas reais despertou o interesse das pessoas desde a década de 70, sendo apontada como a nova interface que ligaria o conteúdo digital ao mundo real, mesmo quando os periféricos utilizados para fazer a RA acontecer eram desconfortáveis, com baixa qualidade e pouca ou nenhuma mobilidade (Caudell, 1992).

Embora tenham ocorrido evoluções em software e hardware que impulsionaram a área da RA, ainda há grandes desafios a serem superados para torná-la massivamente popular. Azuma (2016) destaca a carência de técnicas de rastreamento mais precisas para melhorar qualidade do registro (ato de combinar informações reais com a cena real de forma alinhada), tanto para ambientes internos quanto externos, considerando ainda mudanças de clima e período noturno. Ele ressalta ainda a necessidade de aplicações de RA tornarem-se adaptativas às mudanças físicas do mundo real e às mudanças provocadas pela movimentação do usuário.

O advento e a evolução dos dispositivos móveis deram mais mobilidade para a RA permitindo que o usuário usufrua da tecnologia em outros lugares que não somente em sua mesa. No entanto, mesmo com todo esse avanço tecnológico e com a quantidade de recursos disponíveis, a RA ainda busca maior amadurecimento como tecnologia utilizada no dia a dia das pessoas, sendo considerada pelos usuários como apenas um *gadget* e não como um software indispensável no cotidiano (Langlotz, Grubert, Grasset, 2013).

Alguns dos principais motivos que levam ao desinteresse em aplicações de RA são: a falta de precisão no registro de informações virtuais, a falta de conteúdo, a poluição visual causada por informações sobrepostas, os usuários consideram os mapas mais eficientes, os usuários não consideram as aplicações atuais úteis na prática, e problemas conceituais como marcadores

virtuais flutuando na tela sem relação com a imagem que está sendo mostrada (Grubert, Langlotz, Grasset, 2011).

Um problema em destaque é a falta de precisão no registro (Nazri e Rambli, 2014), especialmente quando o usuário está em um ambiente interno e a aplicação utiliza o GPS (*Global Positioning System*) como tecnologia principal de localização para realizar o rastreamento. Isso porque em curtas distâncias, o GPS disponível nos dispositivos móveis possui baixa acurácia (Valbuena, 2010) principalmente quando existem obstruções entre o dispositivo e os satélites usados para calcular a posição. Essa baixa acurácia interfere diretamente na qualidade do registro e consequentemente na experiência do usuário.

Assim, a presente tese apresenta um modelo adaptativo às transições entre ambientes internos e externos para aplicações de RAM (Realidade Aumentada Móvel). O modelo contém um módulo gerenciador que recebe os dados do ambiente real e elege, de forma automática, a melhor utilização entre técnicas de rastreamento com sensores de longo alcance (GPS), sensores de curto alcance (Wi-Fi, Bluetooth, entre outros), e técnicas de visão computacional, visando mitigar os problemas de localização e orientação.

A adaptação às transições entre ambientes faz com que o usuário utilize a mesma aplicação em diferentes ambientes, fazendo com que o usuário explore POIs (Pontos de Interesse, do inglês *Points of Interest*) internamente e externamente sem a necessidade de instalar duas ou mais aplicações. Além disso, essa adaptação transparente (sem a necessidade de intervenção) para o usuário torna a atividade de exploração mais direta e intuitiva.

A fim de realizar essa adaptação em diferentes níveis (além do interno e externo) foi proposto uma organização hierárquica para a localização do usuário e dos POIs. Dessa forma, tanto a localização do usuário fica mais intuitiva quanto a apresentação de POIs pode ser organizada de forma que os POIs que estão fora do contexto atual do usuário são agregados em um único ícone, reduzindo a oclusão dos marcadores dos POIs.

Para validar a proposta, foi desenvolvida uma aplicação de RAM que se adapta conforme a posição atual do usuário em uma estrutura hierárquica predefinida. A aplicação elege a melhor tecnologia de posicionamento conforme a acurácia necessária o que inclui GPS, localização interna por Wi-Fi e reconhecimento de imagem. Além disso, a aplicação mostra a posição atual do usuário de forma hierárquica e conta com um mini radar que mostra uma visão aérea da

posição atual em relação aos POIs. Adicionalmente, foram implementadas formas de exibir conteúdos multimídia sobre os POI cadastrados.

Com a finalidade de avaliar o modelo através da aplicação, foram conduzidos testes de usabilidade com avaliadores experientes em tecnologia da informação (com pelo menos graduação em curso de ciência da computação ou a afins). A avaliação teve uma etapa de experimentação da aplicação e outra etapa de entrevista sobre a etapa anterior. Como resultados obtidos, foi verificado que a adaptação da aplicação foi vista positivamente pelos participantes, com ressalvas de demora no tempo de atualização da aplicação e acurácia do posicionamento dos marcadores virtuais.

1.1 Justificativa

Os *smartphones* são os dispositivos tecnológicos que têm o maior crescimento de utilização pela população, como mostra a pesquisa realizada pela Fundação Getúlio Vargas de São Paulo (FGV-SP)¹ em 2016. A pesquisa aponta que o uso de *smartphones* no Brasil já é superior ao uso de *notebooks* e *tablets* e até o momento da pesquisa já existiam 1,2 dispositivos portáteis por habitante.

Um dos impactos desse crescimento é a introdução de tecnologias inovadoras como a RA no cotidiano das pessoas, uma vez que a mesma é facilitada nos *smartphones* pelo acúmulo de funcionalidades e sensores que esses dispositivos possuem. Sendo assim, trabalhos que tenham esse escopo de pesquisa podem ter uma evidente importância na forma como essas tecnologias são utilizadas pelas pessoas.

Apesar dos dispositivos móveis trazerem mais mobilidade na utilização das aplicações de RAM, os mesmos possuem limitações (Markov-Vetter, 2013) como pouco espaço em tela para desenho, desempenho computacional limitado quando comparado aos computadores de mesa, consumo de bateria, e a heterogeneidade de hardware e software dos dispositivos. As limitações consequentemente geram desafios na implementação das aplicações de RA.

Outro desafio está relacionado com a insuficiência de conteúdo disponibilizado para interação. Geralmente, as aplicações oferecem seu conteúdo embutido ou possuem um framework privado para inserção de dados (Butchart, 2011). O que acarreta na existência de

¹ <http://eaesp.fgvsp.br/ensinoeconhecimento/centros/cia/pesquisa>

várias aplicações para diferentes propósitos, forçando o usuário a instalar uma aplicação para uma finalidade específica e repetir esse processo toda vez que se move entre ambientes.

O problema da transição entre ambientes torna relevante que as aplicações se adaptem ao contexto dos usuários, forneçam dados sob demanda de acordo com a necessidade e simultaneamente previnam a poluição visual causada pela grande quantidade de informação mostrada em uma tela pequena.

1.2 Objetivos

O objetivo geral deste trabalho é apresentar um modelo adaptativo as transições entre ambientes externos e internos para aplicações de RAM.

Os objetivos específicos são:

- Definir um modelo adaptativo entre ambientes internos e externo para aplicações RAM;
- Desenvolver uma hierarquia de posicionamento do usuário considerando conceitos de Áreas, Pontos e Itens de interesse.
- Mostrar o uso da visão computacional em aplicações de RAM adaptativas (ambientes internos, externos e transição).
- Avaliar o modelo proposto através de cenários reais.
- Melhorar a experiência dos usuários no uso de aplicações de RA de mão para transição entre ambientes interiores e exteriores.

1.3 Contribuições

As principais contribuições desta tese são:

- A elaboração de um modelo de arquitetura para aplicações de RAM adaptativo às transições entre ambientes interiores e exteriores.
- Um *pipeline* híbrido e adaptativo de registro.
- A organização hierárquica da localização dos POIs e do usuário, servindo como base para o detalhamento ou agregação de conteúdo.

1.3.1 Publicações Relacionadas com a Tese

- **SANTOS, C.; ARAÚJO, T.; MORAIS, J.; MEIGUINS, J. Hybrid Approach using sensors, GPS and Vision Based Tracking to Improve the Registration in Mobile Augmented Reality Applications.** International Journal of Multimedia and Ubiquitous Engineering, 2017.
- **SANTOS, C.; MIRANDA, B.; ARAÚJO, T.; CARNEIRO, N.; SOARES, A.; MOTA, M.; MORAIS, J.; MEIGUINS, B. Guidelines for Design Graphical User Interface in Mobile Augmented Reality Application.** Proceedings of HCI International, Toronto, 2016.
- **SANTOS, C.; ARAUJO, T.; SILVA, E.; CHAGAS, P.; MORAIS, J.; MEIGUINS, B. S. Recognizing Azulejos of Texture Pattern: Combining Computer Vision Techniques with Global Positioning System.** Proceedings of XII Workshop de Visão Computacional, Campo Grande, 2016. p. 123-128.
- **SANTOS, C.; ARAUJO, T.; MURAKAMI, B.; BRITO, L.; MORAIS, J.; MEIGUINS, B.; SANTOS, Y. Time and Precision Evaluation of Algorithms to Detect Historical Buildings' Facades.** Proceedings of XII Workshop de Visão Computacional, Campo Grande, 2016. p. 225-230.

1.3.2 Publicações Submetidas

- **SANTOS, C.; ARAÚJO, T.; CHAGAS JUNIOR, P.; MEIGUINS, B; NETO, N.. Recognizing and Exploring Azulejos on Historic Buildings' Facades by Combining Computer Vision and Geolocation in Mobile Augmented Reality Applications.** Journal of Mobile Multimedia, 2017.

1.3.3 Publicações aceitas (em andamento para publicação final)

- **DIVINO, R.; SANTOS, C.; MEIGUINS, B. A Visual Representation of Clusters Characteristics using Edge Bundling for Parallel Coordinates.** In: 21th International Conference Information Visualisation, Londres, 2017.
- **CHAGAS, P.; FREITAS, A.; DAISUKE, R.; MIRANDA, B.; ARAÚJO, T.; SANTOS, C.; MEIGUINS, B.; MORAIS, J. Architecture proposal for data extraction of chart images using Convolutional Neural Network.** In: 21th International Conference Information Visualisation, Londres, 2017.

1.3.4 Publicações Realizadas Durante o Doutorado

- ARAÚJO, T.; SANTOS, C.; MIRANDA, B.; CARNEIRO, N.; SOARES, A.; MOTA, M.; NETO, N.; MEIGUINS, B. **Aspects of Voice Interaction on a Mobile Augmented Reality Application**. Proceedings of HCI International, Toronto, 2016.
- SANTOS, C.; FREITAS, A.; MIRANDA, B.; CARNEIRO, N.; ARAUJO, T.; SOARES, A. ; MORAIS, J.; MEIGUINS, B. S. **A Low Level Evaluation of Head-Tracker and Speech Commands Interactions in Information Visualization Tasks**. In: 20th International Conference Information Visualisation, Lisboa, 2016. p. 159-164.
- SOARES, A.; SANTOS, C.; MENDONCA, S.; CARNEIRO, N.; MIRANDA, B.; ARAUJO, T.; FREITAS, A.; MORAIS, J.; MEIGUINS, B. **A Review of Ways and Strategies on How to Collaborate in Information Visualization Applications**. In: 20th International Conference Information Visualisation (IV), Lisbon. 2016, p. 81-87.
- MIRANDA, B.; CARNEIRO, N.; ARAUJO, T.; SANTOS, C.; FREITAS, A.; MORAIS, J.; MEIGUINS, B. **Categorizing Issues in Mid-air InfoVis Interaction**. In: 20th International Conference Information Visualisation (IV), Lisbon, 2016, p. 242-246.
- MIRANDA, B.; SANTOS, C.; CARNEIRO, N.; ARAÚJO, T.; SOARES, A.; MOTA, M.; NETO, N.; MEIGUINS, B. **Evaluation of Information Visualization Interaction Techniques using Gestures and Widgets in 3D Environments**. Proceedings of HCI International, Toronto. 2016.
- FREITAS, A.; SANTOS, C.; Mota, M.; MEIGUINS, B. **Sugestões para projetos de interface que usam rastreamento de cabeça e comandos de voz**. In: XV Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais, São Paulo, 2016.

1.4 Organização do Trabalho

Este trabalho está organizado em cinco capítulos: Introdução, Realidade Aumentada, Modelo de Arquitetura, Protótipo de Validação e Avaliação da Aplicação.

1 Introdução: apresenta o trabalho de maneira geral, contextualizando a área e justificando da relevância do trabalho;

2 Realidade Aumentada: introduz alguns conceitos importantes para o entendimento do trabalho;

3. Modelo de Arquitetura: apresenta o modelo adaptativo às transições entre ambientes internos e externos;

4. Protótipo de Validação: mostra as funcionalidades implementadas na aplicação utilizada para validar modelo de arquitetura;

5. Avaliação da Aplicação: expõe as avaliações usabilidade realizadas com avaliadores experientes, assim como seus resultados.

2. REALIDADE AUMENTADA

Este capítulo descreve o contexto em que este trabalho está inserido, apresentando conceitos importantes para o entendimento desta tese.

2.1 Conceitos e Teoria

A RA é uma técnica utilizada para aumentar os sentidos do ser humano, seja adicionando ou completando o mundo real com conteúdo digital usando computadores (Bimber, Raskar, 2005).

O conceito de RA está relacionado ao de Realidade Virtual (RV), que por sua vez, sintetiza um mundo artificial fornecendo interatividade por meio do senso de visão, audição, tato ou outras formas de *feedbacks*. A RA também realiza essa experiência interativa, mas seu foco é complementar o mundo real com elementos virtuais, ao invés de criar um ambiente inteiramente virtual (Bimber, Raskar, 2005).

Há ainda estudos que classificam a RA como uma particularização de uma área mais abrangente, conhecida como Realidade Misturada (RM). Segundo Milgram et al. (1994) a RM é um contínuo entre a RA e a Virtualidade Aumentada (VA) (Figura 1). Sendo que na VA o ambiente virtual é enriquecido com objetos reais, prevalecendo os elementos virtuais, enquanto que na RA o ambiente real é enriquecido com objetos virtuais, prevalecendo os elementos reais.

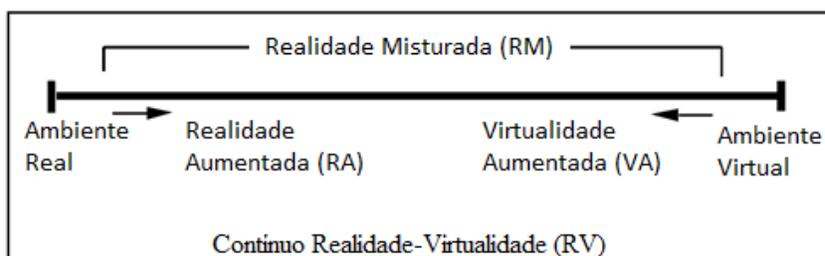


Figura 1. Contínuo Realidade-Virtualidade adaptado de Milgram et al. (1994).

Azuma et al. (2001) definem a RA como um sistema que combina elementos virtuais com cenas do ambiente real, apresenta a combinação para o usuário, é interativa e executa em tempo real, dando a impressão que os objetos virtuais realmente estão no ambiente real.

Existem vários sistemas de RA para plataformas desktops e dispositivos móveis. As aplicações de RA são encontradas em diferentes áreas, como: educação (Wu, 2013), jogos (Alessandra, 2015), física (Duraishwami, 2016), geologia (Donahue, 2016), engenharia (Chi, 2013), medicina (Riddle, 2017), entretenimento (Furt, 2011), turismo (Yovcheva, 2015), entre outras (Chatzopoulos, 2017).

2.1.1 Sistemas de Realidade Aumentada

A RA integra as informações virtuais dentro das cenas do ambiente real ampliando os sentidos e a percepção do usuário. Para melhor entender o funcionamento das aplicações de RA Bimber e Raskar (2005) descreveram um *pipeline* padrão para tais aplicações ilustrado na Figura 2.

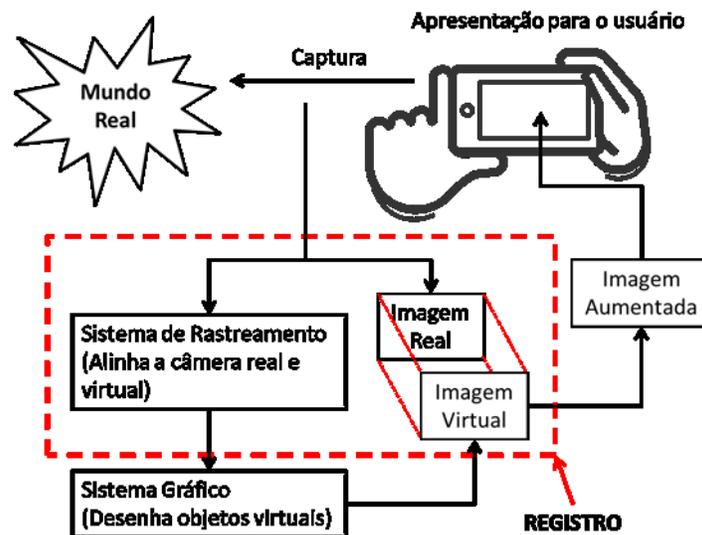


Figura 2. Cenário típico de um ambiente de Realidade Aumentada. Adaptado de (Bimber, Raskar, 2005).

Os componentes do cenário apresentado por Bimber e Raskar (2005) são descritos na lista a seguir:

- **Captura do mundo real:** é o processo de obtenção de dados do mundo real, como imagens, dados de posicionamento e orientação, para calcular informações como a localização do usuário e dos POIs ao seu redor no mundo real;

- **Sistema de Rastreamento:** é responsável por processar as informações brutas obtidas pela captura do mundo real, alinhando as informações virtuais ao mundo real. Existem dois tipos de sistemas de rastreamento:
 - **Rastreamento Óptico:** realiza o reconhecimento de padrões na imagem obtida pela câmera e gera parâmetros de posicionamento para as informações virtuais;
 - **Rastreamento por Sensores:** utiliza os dados capturados por sensores como GPS, bússola, acelerômetro, e giroscópio para calcular os parâmetros de posicionamento das informações virtuais.
- **Sistema gráfico:** é responsável pelo desenho dos objetos virtuais, sendo importante para melhorar o realismo das informações virtuais apresentadas;
- **Registro:** é o ato de combinar camadas virtuais com cenas do ambiente real. Este deve ser feito de tal maneira que os objetos virtuais estejam bem alinhados a cena real;
- **Apresentação para o usuário:** é responsável por mostrar a imagem aumentada para o usuário. Existem vários tipos de tecnologias de apresentação do mundo aumentado, a Figura 3 mostra uma classificação dessas tecnologias de apresentação em relação a sua disposição no espaço e a proximidade ao usuário.

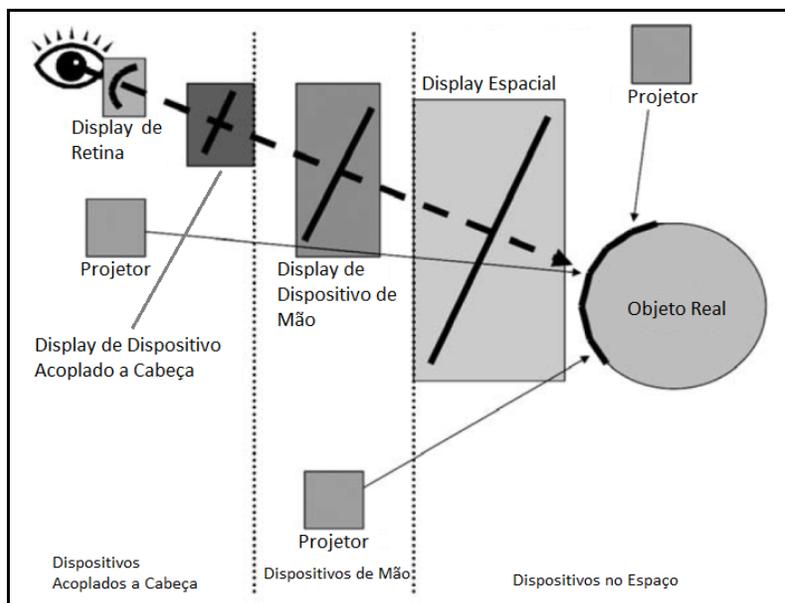


Figura 3. Tipos de *display* pela sua posição. Adaptado de Bimber e Raskar (2005).

Dependendo do tipo de display utilizado, a RA pode ter mais mobilidade, como no caso do display de retina, ou menos como no caso dos displays espaciais. Isso porque os displays espaciais são maiores e mais difíceis de acompanhar o usuário por longas distâncias, enquanto que os displays de retina e displays de mão podem ser carregados pelo usuário para vários locais.

2.1.2 Estilos Arquiteturais em RA

Butchart (2011) realizou um estudo sobre o estilo arquitetural empregado nos navegadores de realidade aumentada e identificou três arquiteturas comumente empregadas, que são: Plataforma, Web e Independente, como mostra a Figura 4. Na arquitetura Plataforma o contexto e os dados são mantidos por uma plataforma, na Web o contexto e os dados ficam espalhados na Internet e na Independente o contexto e os dados ficam embutidos na aplicação.

Dentre as três arquiteturas, a Web (b) é a mais adequada para atender a requisição de escalabilidade dos dados, uma vez que a arquitetura Plataforma (a) requer que uma empresa ou instituição publique o conteúdo da aplicação, assim como a arquitetura Independente (c) precisa que o conteúdo esteja todo embutido na aplicação.

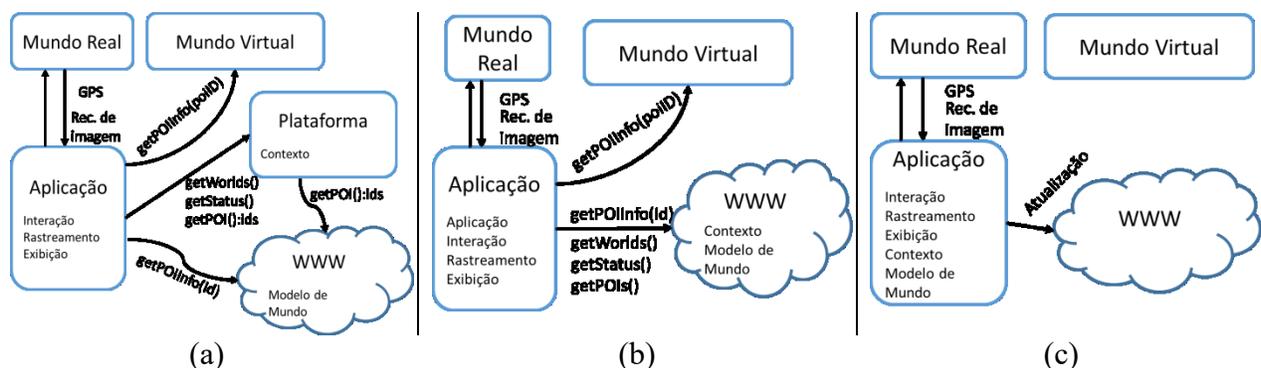


Figura 4. Arquiteturas identificadas por Butchart (2011). As arquiteturas são: Plataforma (a), Web (b), Independente (c).

Sambinelli e Arias (2014, 2015) realizaram uma proposta para padronizar o estilo arquitetural Web, adicionando o núcleo dos navegadores Web atuais (como uma camada interna da aplicação), permitindo que os navegadores de RA pudessem interpretar e executar as mesmas linguagens que os navegadores Web, por exemplo, HTML e Javascript.

Apesar desses esforços em estudar os estilos arquiteturais atuais para aplicações de RA, ainda há uma carência de estudos que diferenciem a arquitetura de RA em relação à arquitetura Web clássica, pois embora existam semelhanças, algumas diferenças que podem ser exploradas para dar mais flexibilidade à aplicação de RA. Por exemplo, enquanto na Web, os serviços estão

disponíveis em detalhes independente da localização do usuário, sendo possível acessar páginas Web que estão em outros continentes com praticamente a mesma facilidade que se acessa uma página Web da rua região. Na navegação de RA é mais interessante que o conteúdo esteja relacionado com as informações mais próximas do usuário, sendo que quanto mais próximo o usuário está mais conteúdo detalhado é possível mostrar.

2.1.3 Realidade Aumentada Móvel

Na RAM a cena do mundo real é capturada pela câmera do dispositivo móvel e é associada a uma camada de conteúdo virtual, que é apresentado na tela do próprio dispositivo. Então, a aplicação apresenta ao usuário a camada virtual sobreposta à camada real, como se fosse uma única camada. Como a captura da cena real e a apresentação ao usuário estão no mesmo dispositivo, o usuário tem a sensação de enxergar através de uma “lente mágica”, uma vez que elementos ou informações que não estavam na cena real, agora podem ser vistas através da tela do dispositivo móvel.

Além da mobilidade dos dispositivos móveis, os sistemas presentes na RA estão agregados em um único dispositivo, bastando apenas um smartphone ou *tablet* para usar aplicações de RA. A lista abaixo descreve os principais desafios no desenvolvimento de aplicações RAM (Martínez et al., 2014) (Wasserman, 2010):

- Integração com sensores dos dispositivos;
- Baixa precisão das tecnologias de rastreamento;
- Restrições e diferentes características físicas dos dispositivos;
- Variabilidade na interface do usuário;
- Falta de padrões de desenvolvimento;
- Consumo de energia.

A lista abaixo descreve algumas diretrizes devem ser consideradas para o desenvolvimento de aplicações RAM, são elas (Dünser, et al., 2007) (De Sá e Churchill, 2013):

- Levar em consideração o perfil do usuário alvo, o uso externo, o uso de uma ou duas mãos, e o tempo de uso da aplicação;

- Seguir os bons princípios de usabilidade para RA e aplicações móveis: cena aumentada limpa, ícones e textos grandes e em camadas, interação com objetos 3D, e considerar o mundo real;
- Levar em consideração as restrições dos dispositivos: reflexão da tela e a falta de precisão no rastreamento;
- Percepção e a cognição do usuário devem ser estimuladas;
- Apresentação das informações virtuais: considerar a quantidade de informação, representação da informação, local das informações, utilização de múltiplas visões.

2.2 Algoritmos de Registro

Existem diversas formas de obter informações do mundo real e também de apresentar o conteúdo aumentado para o usuário. Contudo, esta tese tem o escopo de aplicações utilizadas em dispositivos móveis, dessa forma serão considerados os registros que levam em consideração unicamente estes dispositivos e o que seja possível realizar com o mesmo sem a combinação com dispositivos que o usuário possa ter, como pulseiras, relógios, óculos, etc.

Portanto, em dispositivos móveis são comumente utilizadas duas formas de registro: o registro baseado em sensores e o registro baseado em visão. A seguir são apresentados os trabalhos representativos dos avanços algorítmicos alcançados nesses dois tipos de registro.

2.2.1 Registro Baseado em Visão

A abordagem de registro baseada em visão é a que mais possui algoritmos publicados na literatura (Billinghurst, Clark, Lee, 2015) (Marchand, Uchiyama, Spindler, 2016), envolvendo vários tópicos nas disciplinas de processamento de imagens e visão computacional. Muitas soluções têm sido propostas para melhorar o desempenho computacional e a acurácia do registro baseado em visão, especialmente para utilização em dispositivos móveis, que geralmente têm menor poder computacional quando comparado aos computadores de mesa.

Os algoritmos de registro em aplicações de RA podem ser divididos em três grupos, de acordo com os dados disponíveis para realizar o reconhecimento. São eles: (a) quando se tem os modelos 3D e texturas dos objetos a serem reconhecidos, (b) quando os objetos são rastreados em tempo de execução e (c) quando se considera que os objetos rastreados são planos e tem-se apenas uma imagem de cada objeto.

O caso (a) com modelos 3D conhecidos são menos comuns em aplicações de RAM, pois nessa abordagem é necessário conhecer previamente as medidas dos objetos a serem reconhecidos, assim como realizar uma etapa de calibração com a câmera. Fazendo com que essa abordagem tenha mais aplicações na robótica. Neste caso, o registro pode ser feito através de um conjunto de correspondências entre os pontos 3D no modelo conhecido e os pontos 2D da imagem gerada pela captura da câmera, viabilizando o cálculo das transformações (translação, escala, rotação e projeção) realizadas na câmera.

A quantidade de correspondências utilizadas para resolver o problema pode ser mínima (Kneip, Scaramuzza, Siegwart, 2011), sendo que três correspondências são suficientes para este caso. Masselli e Zell (2014) melhoraram o tempo de execução do algoritmo e sua robustez a ruídos utilizando novos parâmetros geométricos para o cálculo da posição da câmera. Wu (2015) propôs a utilização de mais uma coordenada, necessitando de 3,5 correspondências, deixando o método mais rápido e sem a necessidade de calibração da câmera.

Quando se utiliza mais correspondências o reconhecimento da imagem fica mais robusto, entretanto demanda mais tempo computacional. O algoritmo clássico que resolve este problema utiliza uma abordagem iterativa para estimar a posição da câmera através de projeções ortográficas e escalas, este é chamado de POSIT (Dementhon e Davis, 1995). As propostas atuais resolvem o problema em tempo linear com relação ao número de correspondências (Kneip, Li, Seo, 2014) (Zheng et al., 2013), inclusive possibilitando o uso de múltiplas câmeras (Kneip, Furgale, Siegwart, 2013). A Figura 5 mostra o algoritmo proposto por Zheng et al. (2013) reconhecendo uma caixa e realizando o desenho tridimensional de sua borda.



Figura 5. Funcionamento do algoritmo proposto por Zheng et al. (2013). A esquerda é mostrada a imagem e pontos utilizados como referência e à direita é mostrado o a saída do algoritmo para uma foto.

O caso (b) em que os objetos são rastreados em tempo de execução tem mais aplicações práticas para aplicações de RAM, uma vez que a cena 3D é identificada a cada momento. Embora essa abordagem funcione bem sem um modelo previamente conhecido, a mesma possui a desvantagem de não reconhecer (distinguir) os objetos construídos em tempo de execução, sendo necessária a combinação com outra técnica caso seja necessário reconhecer objetos.

O cálculo eficiente da posição da câmera é conhecido como SLAM (*Simultaneous Localization And Mapping*), pois a ideia é localizar a posição da câmera ao mesmo tempo em que se mapeia o seu movimento entre frames de um vídeo (Marchand, Uchiyama, Spindler, 2016). É importante observar que a abordagem anterior pode ser feita em apenas uma imagem, enquanto que essa abordagem necessita de uma sequência de imagens para ser realizada.

Os primeiros algoritmos computacionalmente viáveis de SLAM utilizam o filtro estendido de Kalman (EKF – *Extended Kalman Filter*) (Eade, Drummond, 2006) (Kang et al., 2010), que é uma versão não linear do filtro de Kalman, utilizado para estimativa de parâmetros ao longo do tempo com variância mínima (Simon, 2010). Uma vez que os algoritmos de SLAM detectam pontos espalhados no espaço tridimensional é possível utilizar um método robusto de estimativa de modelos para encontrar objetos, como planos (Chekhlov et al., 2007).

Uma alternativa promissora para o uso de EKF é o ajustamento de grupo (BA – *Bundle Adjustment*), que otimiza os parâmetros da câmera e as configurações do ambiente de forma iterativa utilizando todos os pontos 3D encontrados na cena. Diferente da abordagem de filtros, o BA pode ser aplicado de forma esparsa em *frames* chaves (Triggs et al., 2000) (Mouragnon et al., 2006), o que, apesar de demandar mais tempo computacional, o torna mais eficiente e preciso (Strasdat, Montiel, Davison, 2010).

Klein e Murray (2007) propuseram uma implementação do SLAM visando os cenários de RA e utilizando a abordagem de BA. Nessa implementação, os autores desenvolveram o rastreamento e o mapeamento ocorrendo em paralelo com o uso de threads, por isso os autores deram o nome de PTAM (*Parallel Traking And Mapping*). Klein e Murray (2009) também mostraram o algoritmo funcionando em um smartphone. A Figura 6 mostra o resultado final (cima) do algoritmo PTAM (Klein, Murray, 2009) em uma sequência de frames chaves de um vídeo (baixo).

O caso (c), em que a posição da câmera é calculada através de um plano conhecido, é o cenário mais utilizado em aplicações de RAM, principalmente porque esta é uma abordagem simplificada, dado que a movimentação de um plano 3D projetado pode ser descrito por uma matriz de transformação projetiva (ou homográfica) 2D (Simon, Fitzgibbon, Zisserman, 2000) e que o modelo utilizado para o reconhecimento é apenas uma imagem.

Esta abordagem consiste na detecção de uma imagem de referência em uma imagem de busca, geralmente a imagem da cena capturada pela câmera. Assim, dado um conjunto de correspondências entre a imagem de referência e a imagem de busca, a posição da câmera na imagem de busca pode ser calculada pela Transformação Linear Direta (DLT – Direct Linear Transform) (Hartkey e Zisserman, 2004).

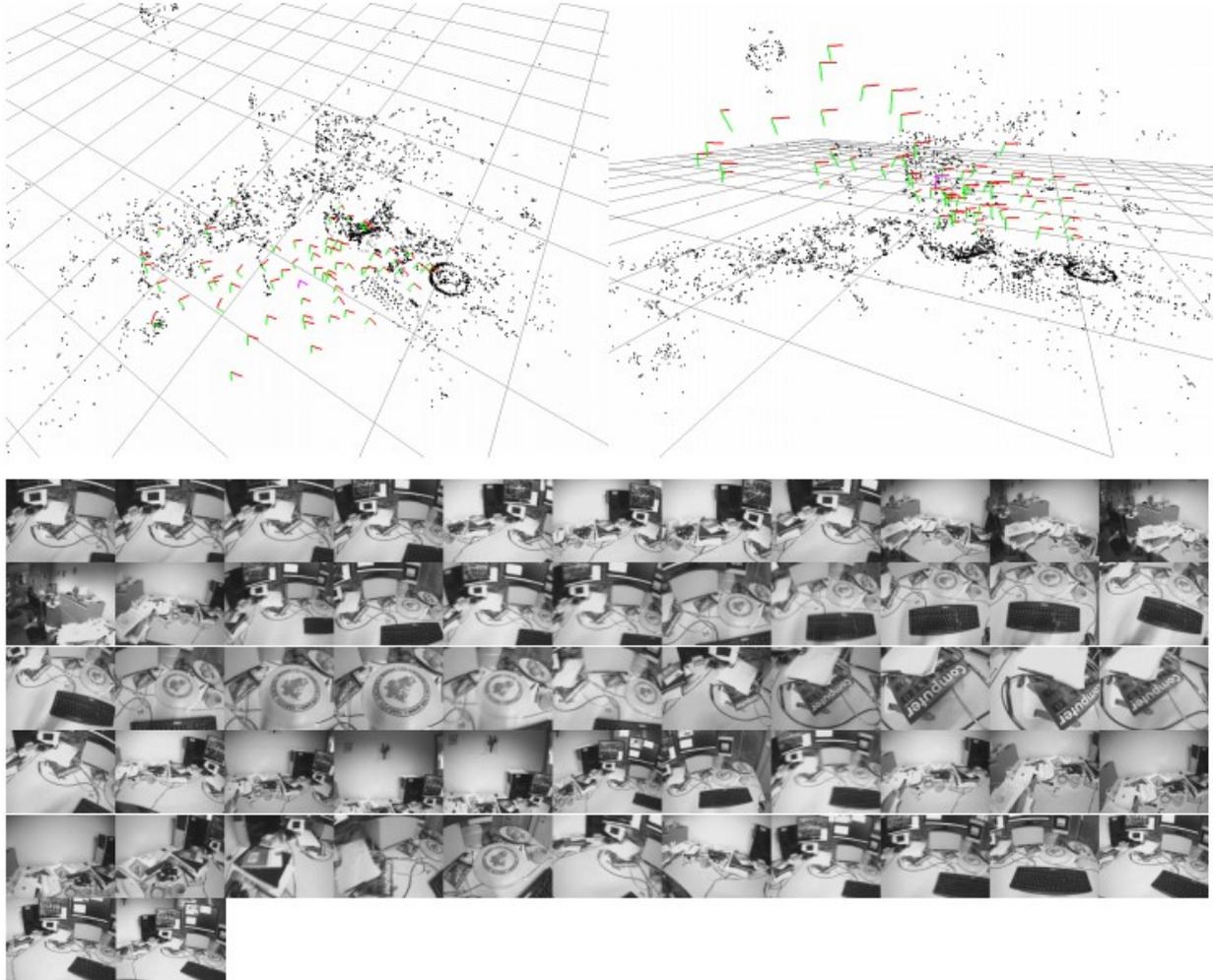


Figura 6. Duas visões do resultado final do algoritmo PTAM (Klein, Murray, 2009) (cima) aplicado a uma sequência de imagens de um vídeo (baixo).

A DLT gera resultados insatisfatórios quando existem correspondências incorretas ou mal posicionadas, que podem ser causadas por ruídos na imagem, inclinações íngremes, entre outros. Dessa forma, geralmente se utiliza um algoritmo robusto baseado na DLT, como é caso do RANSAC (Fischler, Bolles, 1981) (Wang, Luo, 2015), LMEDS (Marchand, Uchiyama, Spindler, 2016), ou M-estimator (Marchand, Uchiyama, Spindler, 2016). Com o uso de planos é possível tanto derivar a posição 3D do objeto (Hartkey e Zisserman, 2004), quanto trabalhar com múltiplos planos (Simon, Berger, 2002).

Várias aplicações de RA foram propostas utilizando essa abordagem, como exemplos podem-se destacar: livros interativos com conteúdo aumentado (Kim, Lepetit, Woo. 2010) (Cho et al., 2011), visitaç o aumentada em museus (Tillon et al., 2010) (Puyuelo et al., 2013) (Miyashita et

al., 2008) (Chen, Chang, Huang, 2014), visualização de projetos e objetos 3D (Dong et al. 2013) (Fuji, Mitsukura, Moriya, 2012) (Kim, Park, Woo, 2014) (Fernández-Palacios et al., 2015), entre outros. A Figura 7 mostra a aplicação de Dong et al. (2013) para a visualização de um terreno utilizando um objeto plano para o reconhecimento.



Figura 7. Aplicação proposta por Dong et al. (2013). As figuras acima mostram o objeto plano que está sendo reconhecido e um eixo virtual. As figuras abaixo mostram a aplicação funcionando.

2.2.2 Registro Baseado em Sensores

O registro baseado em sensores possui duas abordagens em dispositivos móveis. São eles: os sensores inerciais (IMU – *Inertial Measurement Unit*) e o GPS.

Existem basicamente três tipos de sensores IMU: o acelerômetro que mede a aceleração aplicada no dispositivo (inclusive a gravidade), o giroscópio que mede velocidade angular do dispositivo e o magnetômetro que mede a direção dos polos terrestre através do campo magnético da terra.

Ao combinar estes três sensores é possível medir a orientação do dispositivo, ou seja, tanto a angulação em que o dispositivo está rotacionado, quanto sua orientação em relação ao norte da terra. Enquanto que com o GPS é possível descobrir a localização do usuário no globo terrestre. Assim, sabendo a posição do usuário, a posição dos POIs (previamente cadastrados ou baixados

de um servidor) e direção em que a câmera do usuário está apontando é possível saber quais POIs devem ser renderizados na tela, para que os mesmos coincidam com a posição real dos POIs.

Os sensores IMU quando utilizado isoladamente não são bons para a realização do registro em aplicações de RAM (Billinghurst, Clark, Lee, 2015), uma vez que estes sofrem com o problema do deslocamento (*drift*) do mundo virtual em relação ao mundo real. Esse problema é mitigado quando se utiliza o GPS para reposicionar o usuário e os pontos de interesse. Mas quando se faz uso dessa combinação (GPS + IMU) o registro se limita aos POIs que estão longe do usuário (Grubert, Grasset, 2013), por conta da acurácia do GPS nos dispositivos móveis. Por esse motivo, as aplicações de RAM que realizam o registro combinando GPS+IMU têm melhor utilização em ambientes externos.

Exemplos de aplicações publicadas com essa abordagem estão na visualização do transporte público (Nurminen, Järvi, Lehtonen, 2014) (Brata, Liang, Pramono, 2015), observação astronômica (Tian et al. 2014) (Tian et al. 2013), treinamento militar (Zhu et al., 2015), e uma grande quantidade de aplicações para descoberta e localização de pontos turísticos (Geiger et al., 2014) (Kim et al., 2016) (Liao, Humphreys, 2015) (Lee et al., 2013) (Lee, Billinghurst, 2013) (Lee et al., 2012). A Figura 8 mostra a aplicação proposta por Brata, Liang e Pramono (2015) para visualização do transporte público.

Poucos trabalhos têm se dedicado à melhoria da apresentação e visualização dos elementos virtuais registrados. Halik (2012) propôs a utilização de variáveis visuais (comumente usadas em técnicas de visualização da informação) para melhorar a apresentação dos marcadores virtuais, com ênfase para a representação da distância pelo tamanho do marcador. Pryss et al. (2016) utiliza a mesma ideia, mas adiciona um algoritmo de clusterização para minimizar a poluição visual quando há muitos POIs na mesma direção. Carmo et al. (2016) utilizam técnicas de agregação e categorização com base nas preferências do usuário (Figura 9), e ainda no mesmo trabalho os autores concluem que os usuário preferem deixar todas as melhorias propostas ativas e que ainda há espaço para mais pesquisas nessa mesma direção.



Figura 8. Aplicação proposta por Brata, Liang e Pramono (2015). Os marcadores apontam a posição do ônibus (verde) e parada (vermelho). À esquerda-acima é mostrado um radar.

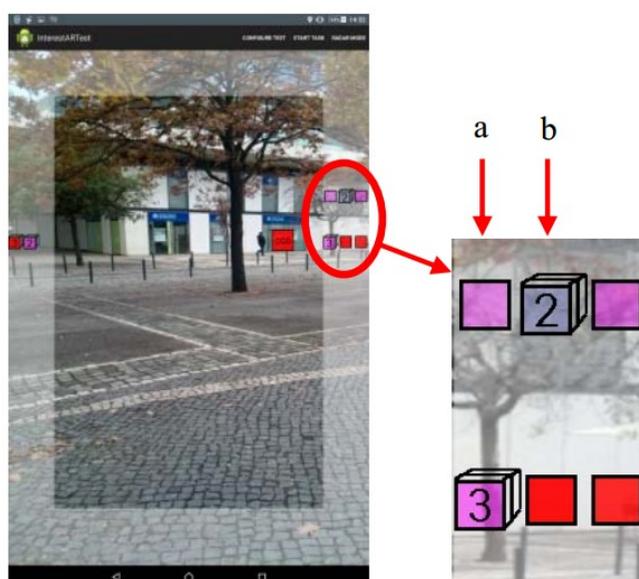


Figura 9. Aplicação proposta por Carmo et al. (2016) à esquerda. À direita é mostrada com detalhes a técnica de agregação proposta.

2.2.3 Registro Híbrido

O registro híbrido consiste na combinação do registro baseado em visão com o registro baseado em sensores, com o propósito de que uma abordagem cubra os pontos fracos da outra. Por exemplo, sabendo que os algoritmos baseados em visão tem melhor acurácia, entretanto são computacionalmente mais custosos, e que os algoritmos baseados em sensores tem baixa

precisão e são mais rápidos, torna-se interessante combiná-los de forma que os pontos fracos sejam atenuados (Billinghurst, Clark, Lee, 2015).

Como o *pipeline* dos algoritmos de visão computacional possuem mais etapas e são mais complexos, os trabalhos científicos usam os sensores para melhorar alguma etapa do *pipeline*, como na detecção das características da imagem (Kurz et al., 2014), na descrição das características (Kurz et al., 2014) (Kurz, Benhimane, 2011a) (Kurz, Benhimane, 2011b) (Kurz, Benhimane, 2012a) (Kurz, Benhimane, 2012b), na criação de restrições para o cálculo da posição da câmera diminuindo a quantidade de iterações (Sweeney et al., 2015) e na fusão dos resultados obtidos pelos registros baseados em visão e sensores individualmente (Yang et al., 2015). A Figura 10 mostra a utilização da gravidade para melhorar os descritores de características da imagem (direita), comparada a descrição baseada somente em visão (esquerda), proposta de Kurz e Benhimane (2012a).



Figura 10. Ao invés de calcular a orientação das características da imagem baseado somente na intensidade dos pixels (esquerda), Kurz e Benhimane (2012a) propuseram a utilização da gravidade para obter a orientação (direita). Abaixo o aplicativo Junaio é mostrado com a proposta aplicada (baixo).

Outras formas de combinação objetivam encontrar o ângulo de um plano em relação ao piso (Jia, Gallagher, Chen, 2013), realizar o rastreamento de objetos utilizando bordas verticais

presentes na imagem (Williams, Green, Billingham, 2013), utilização de um marcador mais simples e atrativo (Li et al. 2016) e melhoria de algoritmos para busca de imagens em grandes bases de dados (Chen, Cao, Wang, 2015).

2.3 Interação em aplicações de RAM

A interação em aplicações de RAM pode ser dividida entre a interação com elementos WIMP (*Window, Icon, Menu, Pointing device*) e interação com elementos virtuais presentes no mundo misturado. Foram abordadas somente as interações que ocorrem com elementos virtuais, pois as interações com elementos WIMP são comuns às mais diversas aplicações, independentemente se a mesma possui funções de RA ou não.

As interações com elementos virtuais podem ser divididas entre interações inseridas no mundo misturado e interações associadas ao dispositivo (Chowdhury et al., 2013). Sendo as interações inseridas no mundo misturado aquelas em que o usuário realiza os gestos dentro da área misturada. No caso dos dispositivos móveis de mão, essas interações ocorrem depois da câmera (Figura 11 (a)). As interações associadas ao dispositivo ocorrem com manuseio direto no dispositivo (Figura 11 (b)), seja tocando, movendo ou pressionando teclas.

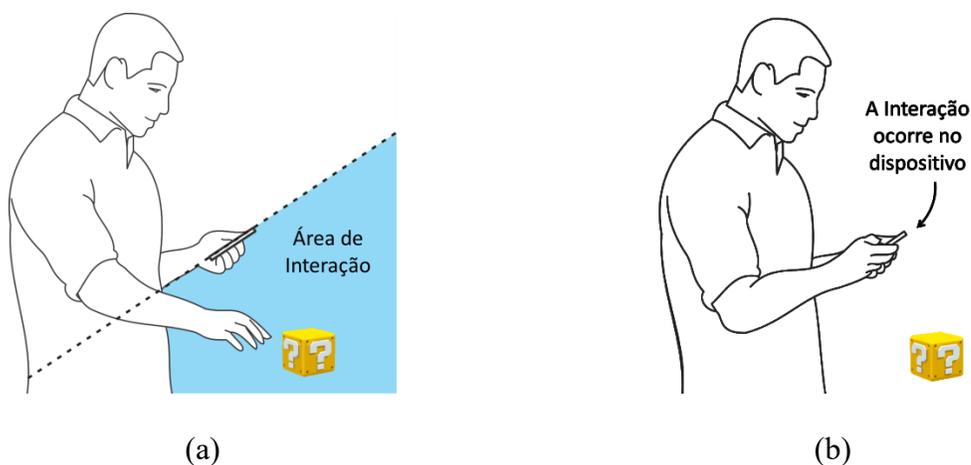


Figura 11. Na interação inserida no mundo misturado as ações ocorrem depois do dispositivo de mão (a). Na interação associada ao dispositivo as ações ocorrem manipulando o dispositivo (b).

A Figura 12 mostra a taxonomia hierárquica de interações com dispositivos móveis proposta a título de organização.

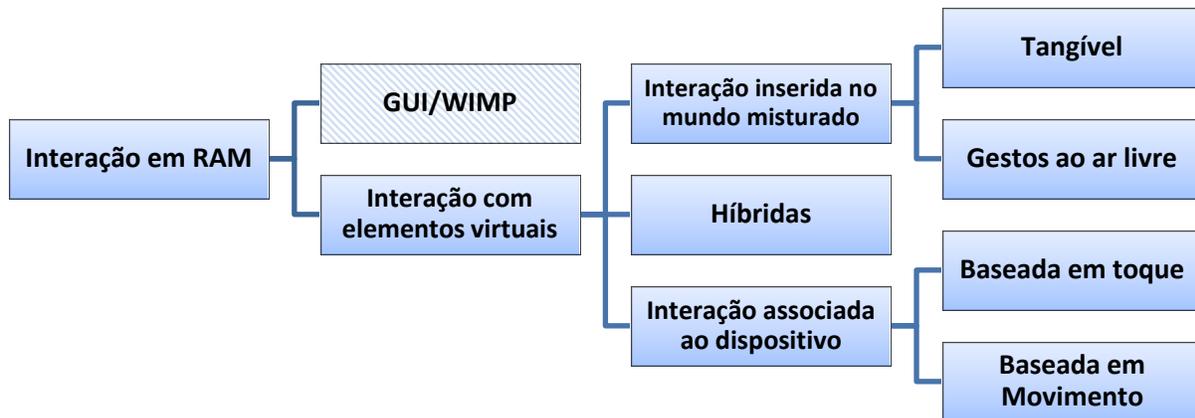


Figura 12. Taxonomia proposta para a organização dos gestos encontrados na literatura.

Outros autores podem classificar as interações de forma diferente (Ahmed et al. 2015), mas essa classificação leva em consideração somente os dispositivos móveis, dessa forma a taxonomia geral foi simplificada para melhor dividir as interações nesse ambiente.

2.3.1 Interações Inseridas no Mundo Misturado

O fato da interação ser inserida no mundo misturado implica que o usuário deve utilizar uma das mãos para interagir com o ambiente, enquanto que a outra mão é usada para segurar o dispositivo, quando utilizada com dispositivos de mão. Sendo essa interação mais confortável quando utilizada com dispositivos vestíveis, pois as duas mãos do usuário ficam livres para realizar interações.

Essa interação ainda pode ser dividida em duas formas, sendo que na primeira, a interação ocorre pela movimentação de objetos físicos (tangíveis), e na segunda, através do reconhecimento de gestos feitos no ar (*mid-air gestures*).

A interação tangível pode ser realizada com a ajuda de acessórios como varetas (Seidinger, Grubert, 2016), pequenas caixas cobertas por marcadores (Neale et al., 2011), ou cartões com várias opções de interação (Tuli, Mantri, 2015) (Billinghurst, Kato, Myojin, 2009) (Carmo et al. 2007). A Figura 13 mostra a aplicação de modelagem 3D para não profissionais proposta por Seidinger e Grubert (2016).

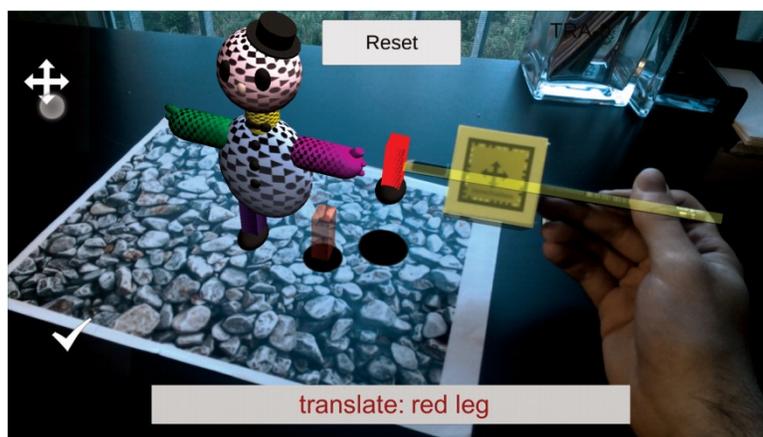


Figura 13. Aplicação MarkerPen (Seidinger, Grubert, 2016) que utiliza uma caneta e um marcador fiducial para realizar transformações (translação, escala e rotação) em avatares 3D.

A interação feita através de gestos no ar é realizada com reconhecimento da mão do usuário (Seo et al. 2008) e pode se estender para reconhecer as pontas dos dedos (Prakash, Kannammal, 2016) (Bai et al. 2013) (Bai, Gao, Billingham, 2013) (Kato, Kato, 2011), que inclusive pode ser utilizada para interação com menus (Choi et al. 2011), ou para realizar operações de transformação (Hürst, van Wezel, 2013) (Lee, Höllerer, 2007). Ainda há a possibilidade de unir as interações gestuais com o uso de marcadores tangíveis (Chun, Höllerer, 2013). A Figura 14 mostra a aplicação proposta por Choi et al. (2011) reconhecendo o gesto feito pelo usuário para interação com menus.

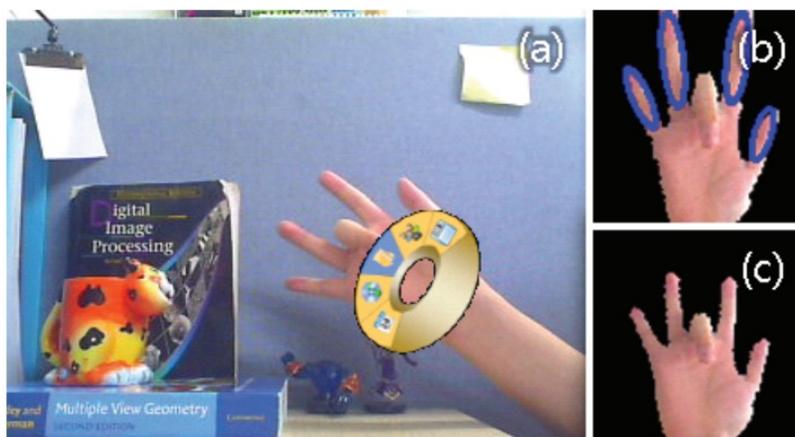


Figura 14. Aplicação de Choi et al. (2011) reconhecendo o gesto feito com a mão para selecionar um item de um menu (a). À direita da figura é mostrada a mão segmentada da imagem original (c) e o resultado da identificação dos dedos (b).

2.3.2 Interações associadas ao dispositivo

As interações associadas ao dispositivo ocorrem devido às ações realizadas no dispositivo, como a movimentação e o toque na tela. Essas interações mitigam a desvantagem do usuário segurar o dispositivo com apenas uma mão (como é o caso das interações inseridas no mundo misturado) e dependendo do tamanho do dispositivo o toque pode ser realizado em toda (ou quase toda) a tela sem que o usuário tenha que tirar a mão do dispositivo.

Dentre as interações que envolvem o toque na tela a mais simples é o toque rápido análogo ao clique do mouse. Essa interação é mais simples, pois o objeto virtual já está registrado na tela e assim basta verificar se o usuário tocou dentro do objeto projetado em 2D. A desvantagem dessa abordagem é a falta de precisão da interação (Ahmed et al. 2015) devido ao dedo obstruir a tela e ocupar uma grande área ao realizar o toque.

Ainda nas interações que envolvem o toque na tela, podem ser realizadas interações gestuais mais complexas, como a movimentação relativa ao arrastar o dedo em qualquer parte da tela (Vincent, Nigay, Kurata, 2013) (Marzo, Bossavie, Hachet, 2014) (Vincent, Nigay, Kurata, 2013b) (Pollmann et al. 2013) e os gestos de múltiplos toques na tela (Telkenaroglu, Capin, 2013). A Figura 15 mostra o resumo visual do trabalho de Pollmann et al. (2013) que realiza uma avaliação comparativa entre as interações de mira fixa e apontador relativo.

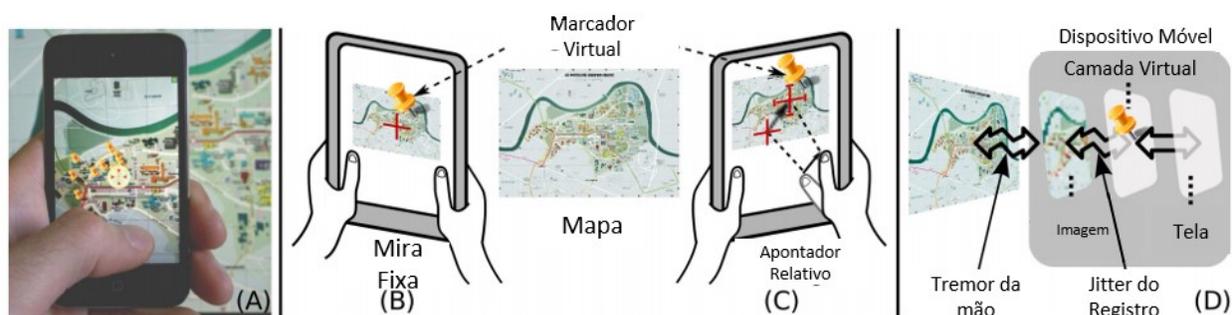


Figura 15. Resumo visual do trabalho de Pollmann et al. (2013) mostrando a aplicação implementada (A), as interações de mira (B), movimentação relativa (c) e o estudo realizado sobre o efeito de desalinhamento que essas interações causam (D).

As interações que envolvem a movimentação do dispositivo são implementadas através de uma mira que tem o objetivo de selecionar um objeto (Hürst, Wezel, 2013) (Hürst, Wezel, 2011) (Vincent, Nigay, Kurata, 2013) (Marzo, Bossavie, Hachet, 2014) (Vincent, Nigay, Kurata, 2013b) (Pollmann et al. 2013), e a partir daí pode-se realizar outras interações, como tocar na

tela para segurar um objeto e realizar a translação do mesmo de acordo com os movimentos do dispositivo (Chowdhury et al., 2013) (Polvi et al., 2016).

As interações em aplicações de RAM geralmente utilizam o toque simples e não possuem uma diversidade de interações possíveis (Langlotz et al., 2014), sendo importante estudar a utilização de diferentes interações para os mais diversos contextos dentro da RAM.

2.4 Aplicações e Frameworks Relacionados

Nesta seção foram selecionados alguns dos principais frameworks e aplicações relacionados com a presente tese. Os frameworks são utilizados para gerar aplicações conforme um padrão pré-estabelecido. As aplicações e frameworks foram selecionados com pesquisas no Google Play Store e foram medidas pelo número de downloads e pelas funcionalidades e recursos que oferecem. A seguir é apresentado um resumo das aplicações e ferramentas selecionadas para comparação.

2.4.1 Awila

O Awila (2014) é uma aplicação de RAM que gera visualizações de POIs com informações extraídas de bases de dados públicas, como o ArcGISOnline, OpenStreetMaps e o GoogleMaps. A escolha da base define o conteúdo que será baixado e carregado em tempo de execução. O uso do mapa pode ser feito com uma projeção na visão da câmera, ou definindo um lugar fixo em que a visualização dele será configurada. A escolha de um POI só é possível enquanto o mapa está projetado na visão da câmera e as informações são mostradas em uma pequena tela que fica a frente do mapa quando um POI é selecionado.

2.4.2 Augment

O Augment (Augment, 2017) é uma aplicação de RAM para reconhecimento e rastreamento de objetos. A aplicação possui marcadores próprios, mas também podem ser definidos pelos usuários. O registro de objetos 3D na cena é feito utilizando esses marcadores, e são oferecidas várias opções de interação ao usuário, como animações, visão separada e troca de paleta de cores.

2.4.3 Junaio

O Junaio (Metaio, 2015) é uma plataforma RA que carrega informações de bases de dados criadas pelo próprio usuário, definido como canal. Esses canais são um conjunto de informações

referentes aos POIs e são carregadas conforme escolha do usuário. O conteúdo de cada POI pode ser visualizado diferentemente de acordo com seu tipo. Quando selecionado, cada POI tem um conjunto de informações e um menu para alguma interação. Existem interações como ligar para o local, definir rota no mapa ou acessar o website do POI, mas essas ações exigem a inicialização de aplicativos externos ao Junaio.

2.4.4 Layar

O Layar (Layar, 2015) é uma aplicação que possibilita que os usuários busquem por POIs através do reconhecimento de imagem, Navegador RA, Mapa, e leitura de QR Code. Neste o usuário pode escolher uma das formas para busca de POIs e ao selecionar um POI ele pode interagir com o conteúdo cadastrado através de páginas. Os divulgadores de conteúdo devem cadastrar suas páginas através de uma plataforma Web que será automaticamente publicada na aplicação.

2.4.5 Blippar

Blippar (Blippar, 2015) é um aplicativo de RAM que usa as funcionalidades de reconhecimento imagem para sobrepor informações virtuais no mundo real capturado pela câmera do dispositivo. Neste os usuários podem executar jogos para celular, vídeos, músicas, e galerias de imagens.

2.4.6 Wikitude

Wikitude (Wikitude Inc., 2015) é um aplicativo RAM inicialmente desenvolvido para prover aos usuários um navegador de RA com base em sensores e GPS, os usuários podiam procurar e explorar POIs através do navegador chamado de Wikitude *World Browser*. Em 2012, a empresa que desenvolve o Wikitude reestruturou o propósito de divulgação do aplicativo, tornando a ferramenta de desenvolvimento a principal forma de comercialização. Agora os provedores de conteúdo podem adquirir o Wikitude SDK (*Software Development Kit*) para disponibilizar seu conteúdo personalizado com as funcionalidades de reconhecimento de imagem, navegador RA, renderização de objetos 3D e vídeos sobrepostos ao conteúdo.

2.4.7 Vuforia

O Vuforia SDK (Vuforia, 2017) é um SDK voltado para reconhecimento de padrões e imagens para registro em RA. Possui várias funções de reconhecimento, como reconhecimento

de objetos pré-definidos, marcadores próprios e reconhecimento de texto. A computação do rastreamento é feita online, enquanto os sensores do dispositivo coordenam parte da movimentação para manter o registro estável.

2.4.8 Comparação

A Tabela 1 possui um resumo da comparação realizada entre as aplicações, frameworks e APIs pesquisadas. A primeira coluna compara as funcionalidades de navegação RA, a segunda relaciona as funcionalidades de mapa, a terceira mostra as características de integração entre as visões (diferentes formas de ver o mesmo conteúdo) e a quarta trata das funcionalidades de reconhecimento de imagens.

Dentre as aplicações pesquisadas apenas três possuem a funcionalidade de navegação de RA (permite ao usuário encontrar POIs geolocalizados), que são: Awila, Layar e Wikitude. O navegador de RA do Awila está integrado com um mapa, mas não possui filtro de distância e nem agregação de POIs quando existem muitos em um mesmo lugar. O Layar possui filtro de distância, mas não possui agregação de POIs. O Wikitude possui o filtro de distância e a agregação de POI por colisão de marcadores, ou seja, quando existem dois ou mais marcadores virtuais colidindo, todos eles são agregados em um único item. Ao selecionar esse item é apresentada uma lista com todos os POIs agregados para o usuário escolher o desejado. O protótipo apresentado nesta tese possui filtro por distância e agregação de hierárquica, que apresenta os POIs dependendo da localidade do usuário e estrutura hierárquica que os POIs estão organizados.

Em relação à navegação por mapa, a aplicação Awila possui a funcionalidade de mapa convencional (contendo interações de *zoom* e *pan*) e um mapa 3D que se move conforme a direção que o usuário está apontando, mas não possui a funcionalidade de rota até o POI. No Wikitude e no Layar as funcionalidades de mapa são feitas em uma aplicação externa. A aplicação proposta possui o mapa convencional, o mapa 3D, dá suporte ao mapa interno dos POIs e possui a funcionalidade de rota integrada na própria aplicação.

A integração entre as funcionalidades (visões coordenadas) das aplicações carece de atenção nas aplicações atuais. As visões coordenadas permitem que o usuário veja os mesmos POIs de diferentes formas e realize as suas tarefas com o método que mais lhe agrade. A aplicação proposta possui integração entre o navegador de RA, o mapa e o reconhecimento de imagem. Outro aspecto importante é a integração entre o ambiente externo e interno, fazendo com que a aplicação reconheça que o usuário chegou ao POI alvo e deseja continuar explorando o conteúdo presente nesse POI. O modelo adaptativo apresentado nesta tese trata deste problema e essa adaptação foi implementada na aplicação proposta que valida o modelo.

O reconhecimento de imagem nas aplicações pesquisadas é realizado para descobrir um POI pré-cadastrado na nuvem, e assim, disponibilizar conteúdo para o usuário com o reconhecimento de imagens naturais (podem ser capas de livros, logomarcas de empresas, pinturas, entre outros). Com exceção da aplicação Awila que não possui acesso a nuvem para descobrir POIs com a câmera. Na aplicação proposta o reconhecimento de imagens é feito de forma integrada com a localização do usuário, levando em consideração a sua posição em relação à hierarquia dos POIs e AOIs. Dessa forma, além dos usuários utilizarem o reconhecimento de imagens para descobrir POIs, essa funcionalidade está integrada à demais visões. Além disso, a aplicação proposta possui um serviço de reconhecimento de cena interno/externo que é utilizado para auxiliar na adaptação aos ambientes externo/interno, quando o GPS falha ao tentar realizar essa tarefa.

3. MODELO DE ARQUITETURA

Este capítulo apresenta o modelo arquitetural desenvolvido para aplicações de RAM adaptativas utilizando uma combinação de dados obtidos por sensores e por algoritmos de visão computacional. Em seguida são descritos os aspectos de implementação deste modelo e como estes foram utilizados na ferramenta que servirá de protótipo de validação para o modelo.

3.1 Descrição do Modelo

As adaptações da aplicação devem ocorrer de forma gradual e transparente para o usuário, sem a necessidade do mesmo trocar de aplicação, ou de janela na mesma aplicação. Para isso, as aplicações de RAM devem interpretar a localização do usuário baseado em informações que o dispositivo móvel pode ter naquele momento, ou classificar cenas e objetos do ambiente através das imagens capturadas.

A Figura 16 mostra o *pipeline* resumido de aplicações de RA com a adição de um módulo de controle adaptativo e a Figura 17 mostra este modelo adaptativo de forma detalhada. O modelo adaptativo está baseado no *pipeline* tradicional de RA (Figura 2) e contém um módulo de Controle Adaptativo que decide quais as estratégias mais apropriadas de localização, registro e apresentação do conteúdo de acordo com o cada ambiente onde o usuário está localizado.

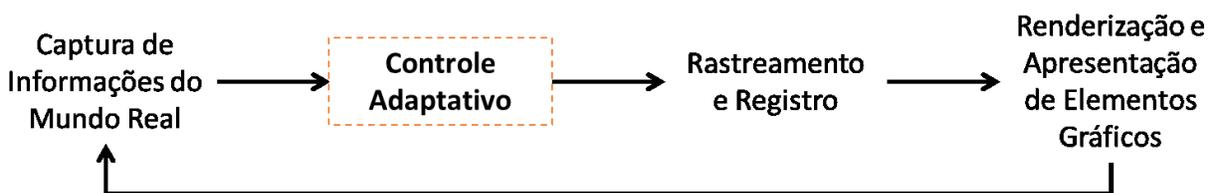


Figura 16. *Pipeline* resumido de RA com a adição de um módulo de controle adaptativo.

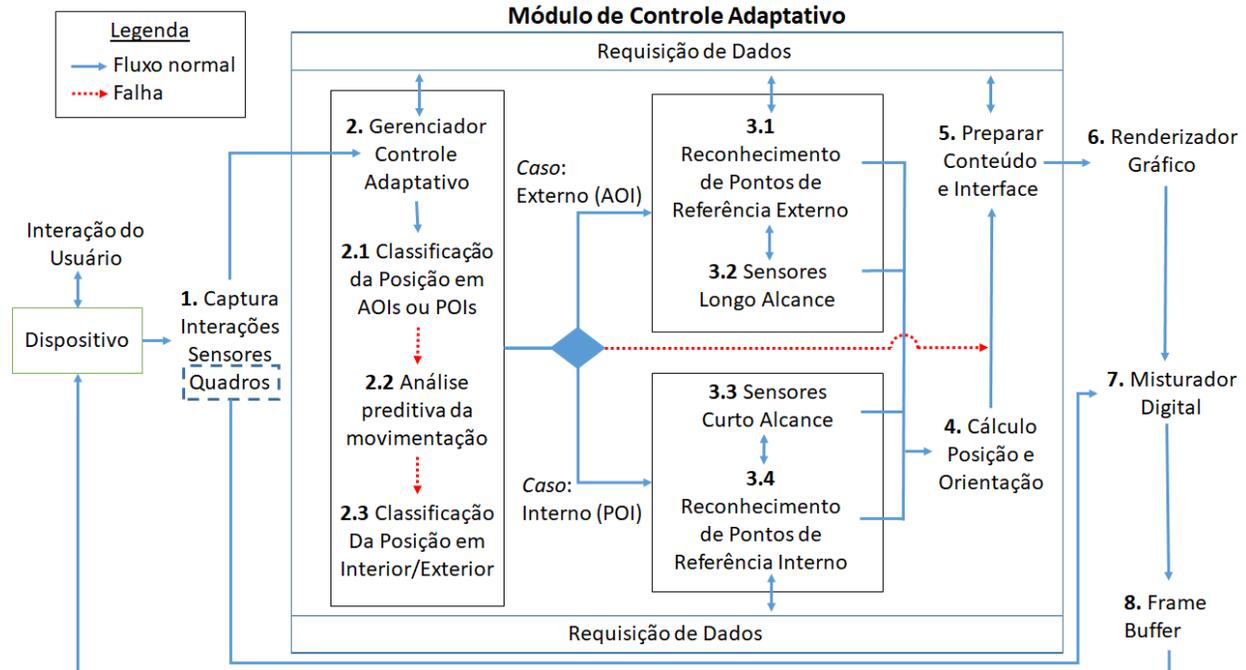


Figura 17. Modelo adaptativo às transições do usuário entre ambientes internos e externos.

A primeira tarefa do Controle Adaptativo é receber e analisar os dados obtidos dos sensores e da câmera do dispositivo móvel (1). A análise realizada pelo Gerenciador de Controle Adaptativo (2) é feita inicialmente para classificar a localização do dispositivo em uma Área de Interesse (AOI) ou um Ponto de Interesse (POI) através de sensores de longo alcance (2.1). Há três situações possíveis de classificações para localização: fora de qualquer AOI (dentro da AOI mundo), dentro de uma AOI, ou dentro de um POI.

A análise de localização é iniciada com a verificação dos dados obtidos pela localização de longa distância em relação às AOIs existentes. Uma AOI é uma área geográfica que pode conter outras AOIs e POIs, e os POIs são locais onde o usuário pode explorar conteúdo. Na leitura dos sensores de posicionamento podem ocorrer falhas ou problemas, como a impossibilidade de leitura gerada por bloqueio do sinal.

Caso a verificação inicial falhe são utilizadas medidas preditivas da movimentação do usuário (2.2), o algoritmo de análise preditiva utiliza medições anteriores para calcular um vetor de direção e velocidade, estimando a posição do usuário quando a informação não está disponível (Kang e Han, 2015) (Kourogi e Kurata, 2014) (Chang et al. 2015). A análise preditiva sofre do problema de erro acumulado, uma vez que se utilizam as medições anteriores para

estimar uma nova posição, e se houver erro nas medições anteriores esse erro é propagado para as próximas medições.

Durante a análise preditiva, o sinal dos sensores de posicionamento pode voltar a ter boa acurácia, neste caso se a medição feita pela análise preditiva estiver longe da posição identificada pelos sensores, essa posição é trocada, mas se a medição não estiver longe é feita uma convergência do posicionamento ao longo do tempo, até que a posição do usuário corresponda à medida pelos sensores de posicionamento.

Caso a análise preditiva esteja atuando a um determinado tempo, de forma que o erro acumulado possa ser grande, são utilizadas (em adição) técnicas de visão computacional para classificar se a imagem capturada pela câmera do dispositivo se trata de um ambiente interno ou externo (2.3), confirmando o posicionamento do usuário.

Uma vez classificada a posição inicial do dispositivo em AOI ou POI, é iniciada a escolha de um método para o cálculo de uma localização mais precisa do dispositivo. Se a classificação for AOI e a precisão da localização externa estiver adequada, os dados desse método de localização são utilizados (3.2), caso contrário, são aplicados algoritmos de visão computacional (3.1), que podem ser calculados no próprio dispositivo ou em nuvem, e uma composição das duas abordagens também é possível (Sensores de longo alcance + Visão Computacional). Se a classificação for um POI, são procurados sensores de curto alcance (3.3), tais como redes Wi-Fi, Beacons Bluetooth, RFID (*Radio-Frequency IDentification*), entre outros, caso contrário, algoritmos de visão computacional são utilizados para inferir a localização do usuário (3.4), novamente é possível a combinação das duas abordagens. Além disso, a orientação da câmera do dispositivo deve ser calculada para a visualização correta do conteúdo aumentado (4), geralmente utilizando os sensores inerciais do dispositivo móvel.

Após o cálculo da localização e orientação do dispositivo móvel são verificados quais conteúdos de AOIs e POIs estão disponíveis para visualização na forma de ícones virtuais, considerando critérios de distância e orientação limite. O conteúdo dos AOIs ou POIs onde o usuário se encontra é obtido através de uma requisição para um provedor de conteúdo (5). Ícones de AOIs podem representar POIs agregados, caso usuário esteja fora da referida AOI, fazendo com que os POIs que estão dentro daquela AOI sejam representados pelo ícone do AOI,

diminuindo a poluição visual da tela do usuário, portanto utiliza-se uma estrutura hierárquica de AOIs.

Caso uma falha ocorra no processo de localização do dispositivo móvel, uma mensagem de feedback é apresentada ao usuário. Dependendo do tipo de falha a aplicação pode solicitar informações que corrijam a mesma ou pode unicamente mostrar uma notificação da falha.

A partir desta etapa, o *pipeline* de RA é seguido de forma tradicional, as AOIs e os POIs visíveis são renderizados (6) e encaminhados ao Misturador Digital (7), que registra o conteúdo virtual com a cena capturada pela câmera do dispositivo. Após a fusão desses dois conteúdos, a imagem aumentada se torna disponível no *framebuffer* (8) do dispositivo para ser apresentado na tela.

Quando os ícones já estão desenhados na tela, o usuário pode realizar uma interação com um ícone virtual no ambiente aumentado, normalmente a ação de seleção é efetuada com um simples toque. Essa ação visa buscar conteúdo associado ao AOI, POI ou IOI selecionado.

3.2 Localização

A localização do usuário é importante para posicioná-lo em relação ao mundo real, viabilizando o posicionamento adequado do conteúdo aumentado, portanto a acurácia da localização tem um impacto direto na qualidade do registro. Conseqüentemente, à medida que o usuário se movimenta é importante decidir qual a melhor estratégia de localização para gerar um registro mais apurado e adequado à situação do usuário e aos dados que o mesmo está visualizando. As próximas seções descrevem como essas estratégias de localização podem ser desenvolvidos para ambientes externos, internos e transição entre eles.

3.2.1 Externa

A estratégia primária de localização proposta passa pela construção de AOIs hierárquicas que são áreas geográficas (círculos, retângulos ou polígonos) que podem conter outras AOIs filhos ou POIs. Os POIs possuem uma AOI especial chamada de *áurea*, que contém os limites da área geográfica do POI, bem como contém os IOIs (Itens de Interesse), que são objetos ou lugares dentro do POI que o usuário pode explorar conteúdo.

Com a posição do usuário obtida por sensores de longo alcance (como o GPS) e a definição das AOIs é possível verificar em qual AOI o usuário está, buscando a AOI de nível mais baixo

na hierarquia. Uma vez que se sabe em qual AOI o usuário está e qual o nível da hierarquia desta AOI, a aplicação pode se adaptar a esse contexto atualizando o conteúdo com nível de detalhamento adequado ao nível da hierarquia da AOI, modificando a representação dos ícones virtuais (tamanho, forma, agregação, etc), se necessário usando algoritmos de visão computacional para POIs próximos, entre outros. Caso o nível mais baixo na hierarquia seja uma área de um POI, este representa o momento em que a estratégia de localização pode ser alterada para dar suporte à localização interna, bem como são carregados os dados dos IOIs para exploração.

A Figura 18 ilustra um exemplo de representação hierárquica das AOIs, POIs e IOIs no mapa e plantas baixas, realizando um *drill-down* na hierarquia da esquerda para a direita. A Figura 18 (a) apresenta um AOI com AOIs filhos, a Figura 18 (b) apresenta o AOI selecionado anteriormente com POIs internos, finalmente a Figura 18 (c) mostra a planta baixa POI selecionado anteriormente. Neste caso, o prédio possui dois andares, sendo necessária uma estratégia de localização interna para definir tanto a localização do usuário quanto o andar que o usuário está.

Quando o usuário está dentro de um AOI são apresentados para o usuário todos os AOIs ou POIs filhos e os AOIs irmãos na hierarquia. Esses AOIs irmão representam de forma agregada todos os POIs presentes neles. Dessa forma, mesmo que o conteúdo da aplicação seja amplo, a sobrecarga de informação na tela do usuário é reduzida melhorando a visibilidade de POIs, uma vez que o espaço em tela dos dispositivos móveis é reduzido.

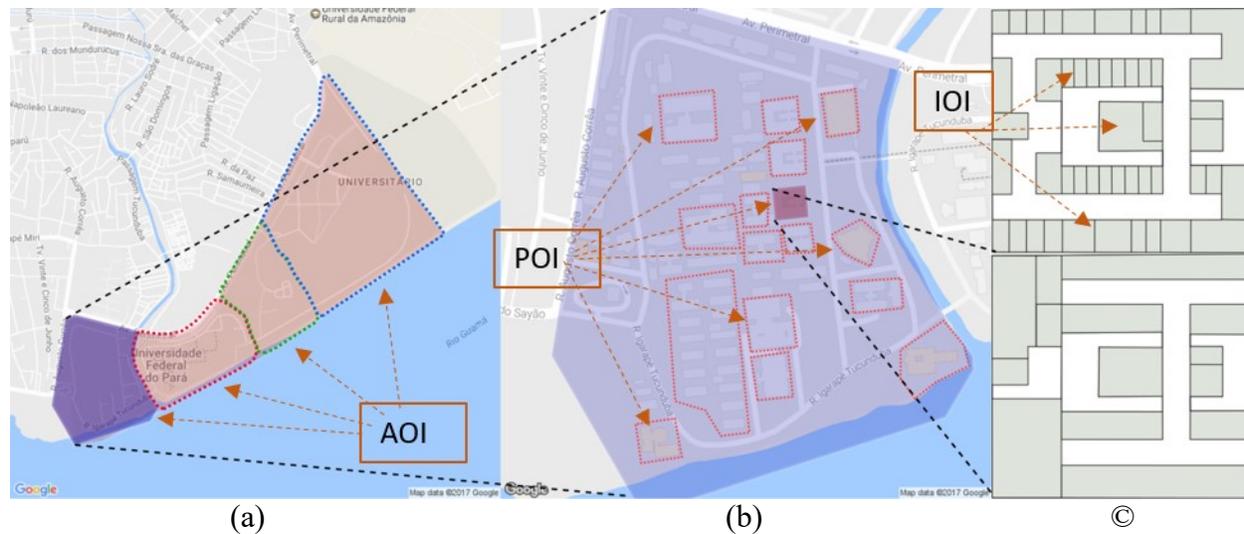


Figura 18. Em (a) é ilustrado dois níveis de AOIs, em (b) é ilustrado o AOI azul de (a) com seus POIs internos, e em (c) é ilustrada a planta baixa dos dois andares presentes no POI vermelho ilustrado em (b).

3.2.2 Transição

Este cenário apresenta alguns desafios, como: o erro presente na localização por GPS que prejudica o registro de POIs próximos ao usuário, alta variabilidade de iluminação e poluição visual que prejudica o registro realizado por algoritmos de visão computacional, e a percepção da mudança de ambiente para adotar nova metodologia de localização o que pode gerar atrasos na adaptação da interface gráfica do usuário influenciando na experiência da utilização.

A grande maioria das aplicações de RAM utiliza GPS e sensores inerciais para determinar a localização do dispositivo e orientação da câmera. Estes funcionam bem quando o POI alvo não está muito próximo ao usuário, mas a curtas distâncias (poucos metros) o GPS não apresenta boa acurácia, prejudicando o processo de registro, e consequentemente a experiência do usuário.

A mudança de ambiente externo-interno, ou interno-externo, pode ser adaptada como base na área do POI, no nível de precisão do GPS ou utilizando algoritmos de visão computacional para classificar ambientes exteriores e interiores. Primeiramente é verificado qual o AOI ou POI mais interno que o usuário está presente através da posição do usuário obtida pelo GPS. Caso o GPS não esteja com uma boa acurácia, algoritmos de reconhecimento de cena interno/externo (Szummer e Picard, 1998), (Serrano et al., 2002), (Kim et al., 2010), (Payne e Singh, 2005) podem ser usados para decidir se o usuário está dentro ou fora de um POI.

Uma vez classificado que o usuário está dentro de um POI, a interface do usuário deve ser modificada para mostrar informações pertinentes para explorar o POI, como exemplo, mostrar o

mapa do local incluindo a posição do usuário e os IOI daquele POI para exploração de conteúdo pelo usuário.

Além da adaptação na interface gráfica e no conteúdo, a aplicação RAM também deve se adaptar aos métodos de localização utilizados, uma vez que em ambientes internos a acurácia do GPS não é suficiente para indicar com exatidão a localização do usuário, especialmente quando o local possui mais de um andar.

3.2.3 Interna

Uma vez classificado em ambiente interno e com a interface já adaptada a aplicação precisa adaptar também o método de localização para registrar os IOIs na aplicação de RAM.

Existem diversas estratégias de localização interna na literatura (Ni, 2004), (Záruba, 2007), (Park, 2010), (Liang, 2016), (Guo, 2016), (Binu, 2016), sendo um método que merece destaque é o uso de sinais de Wi-Fi, uma vez que geralmente os locais internos já possuem uma infraestrutura de rede local que fornece Internet, permitindo a utilização apenas da potência do sinal de cada ponto de acesso retirando a necessidade de adquirir novos equipamentos.

Uma vez que a localização do usuário tem uma boa acurácia, o usuário pode reconhecer e explorar os IOIs presentes no POI. Para reconhecer os IOIs, basta o usuário apontar a câmera para eles e como a aplicação conhece a posição do usuário e dos IOIs só é necessário executar o algoritmo de reconhecimento para os IOIs próximos.

3.3 Aspectos de Implementação

Para descrever os componentes do modelo em nível de implementação foram feitos alguns diagramas UML (*Unified Modeling Language*), incluindo caso de uso, sequências e classes. Os diagramas foram desenvolvidos de forma simplificada para melhorar a leitura e facilitar o entendimento dos mesmos, não sendo apropriados para tradução direta em código fonte. Esses diagramas também representam a estrutura de código seguida na implementação do protótipo de validação do modelo (Capítulo 4).

3.3.1 Diagrama de Caso de Uso

A Figura 19 apresenta o diagrama de caso de uso com as funcionalidades disponíveis no protótipo de validação. O usuário pode realizar cinco atividades com aplicação, são elas: traçar rota até o POI, pesquisar POI, visualizar POI, reconhecer imagem, navegar com RA e navegar

com mapa. O POI pode ser visualizado através de diversas mídias, como imagem, texto, vídeo e outros. Assim como o reconhecimento de imagens pode reconhecer diferentes objetos e marcadores dependendo das informações disponibilizadas para executar o algoritmo de reconhecimento.

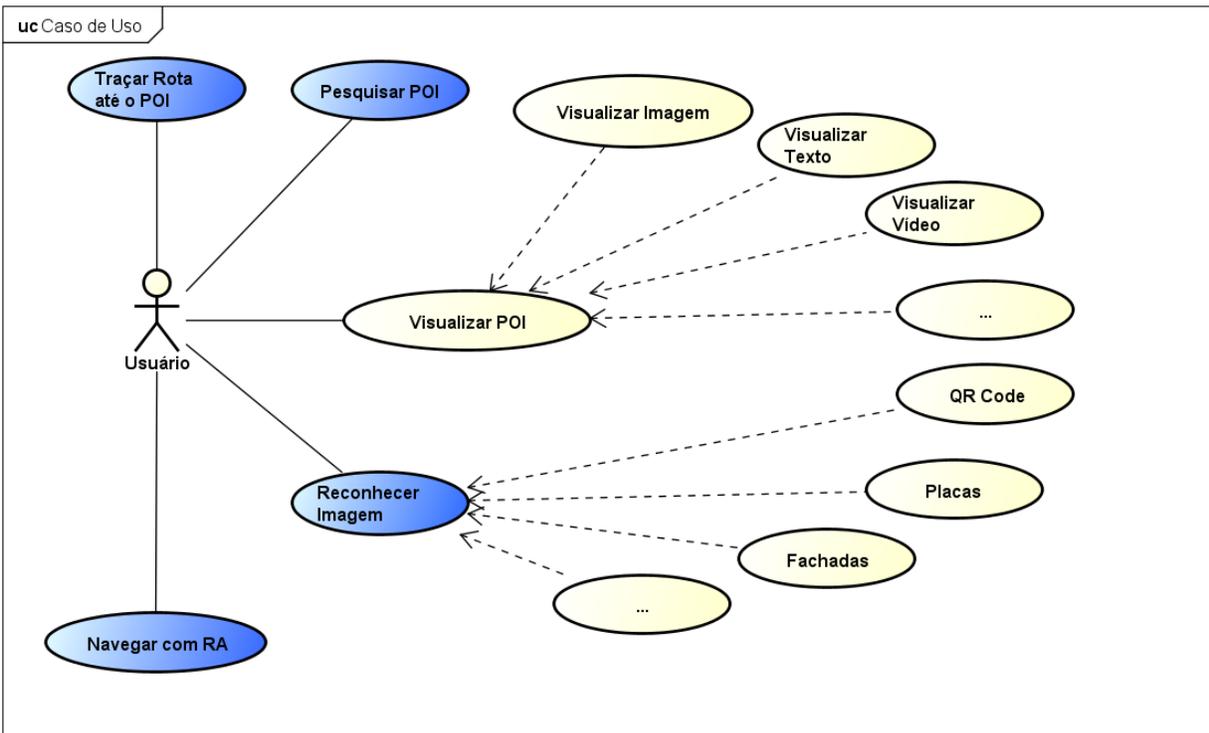


Figura 19. Diagrama de casos de uso. Em azul os casos de uso adaptativos, e os casos de uso descritos com “...” indicam uma sequência de exemplos de casos de uso.

As funcionalidades em azul são adaptativas às transições do usuário, ou seja, se adaptam de acordo com a posição atual do usuário, por exemplo, reconhecer imagem depende da posição atual do usuário e para onde o usuário está apontando sua câmera, assim como, para navegar com RA que agrega marcadores de POIs de acordo com a posição do usuário. Enquanto que as funcionalidades de visualizar POI e pesquisar POI funcionam de forma igual independente da posição atual do usuário.

A funcionalidade de traçar rota depende da posição atual do usuário e é adaptativa durante o trajeto, pois o usuário deve ser levado primeiramente às entradas dos AOIs de forma hierárquica até a entrada do POI e depois o trajeto interno do local.

Os Casos de Uso descritos como “...” indicam que podem existir mais casos de usos do mesmo gênero na aplicação, por exemplo, no caso de visualizar POI este caso significa podem ser visualizadas mais formas além das listadas, como: áudio, conteúdo e animações 3D, entre outros.

3.3.2 Diagrama de Classes

A Figura 20 mostra uma visão geral dos pacotes e classes utilizados para a implementação do modelo adaptativo. A *AdaptiveManagerActivity* é a classe principal do projeto, sendo responsável pela lógica de adaptação e pela comunicação com as demais dependências do projeto. As dependências do sistema utilizam as bibliotecas de aprendizado de máquina e de visão computacional para utilizar modelos de aprendizado de máquina e extração de características em imagens respectivamente.

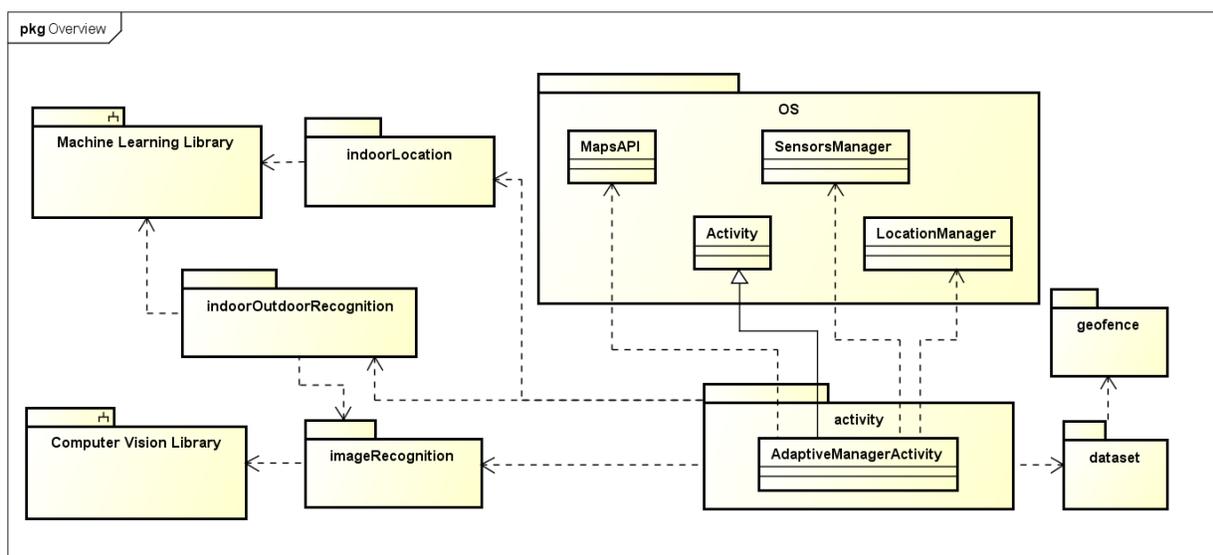


Figura 20. Visão geral dos diagramas de classe.

Abaixo são descritas as funções de cada pacote presente no diagrama:

- **indoorLocation**: responsável por fornecer a localização do usuário em um ambiente interno.
- **indoorOutdoorRecognition**: responsável por realizar a classificação entre ambientes internos e externos dado uma imagem do local.
- **imageRecognition**: responsável por fornecer algoritmos de reconhecimento de imagens.

- **dataset**: responsável por fornecer os dados e mídias sobre os POIs e outros itens que a aplicação pode fornecer para o usuário. Contém interfaces facilitadas para leitura de dados tanto on-line quanto off-line.
- **geofence**: responsável por fornecer uma estrutura de dados hierárquica de regiões geográficas. Possui classes que facilitam a organização e notificação sobre alterações na localização do usuário, independente se a localização é interna ou externa.
- **OS**: possui as classes disponibilizadas nativamente para programação de dispositivos móveis, este fornece serviços de acesso aos sensores dos dispositivos móveis, como GPS, acelerômetro, Wi-Fi, Bluetooth, entre outros.

A Figura 21 mostra as classes e pacotes envolvidos na localização do usuário usando áreas geográficas que delimitam uma região, chamadas de *Geofence*. Os AOIs, POIs e IOIs são implementados na mesma classe, chamada de POI. Essa classe possui um pai do mesmo tipo, dando a organização hierárquica. Cada POI possui um *Geofence* que delimita a sua área, auxiliando na verificação da posição do usuário em relação aos AOIs, POIs e IOIs.

A classe *LocationManager* pertence ao pacote OS (nativo do sistema operacional) e é responsável por fornecer a localização do usuário obtida por sensores de longa distância. Para receber as notificações de alterações da posição do usuário é necessário implementar uma interface de retorno das informações, chamada de *LocationListener*. Os POIs são gerenciados pelo *POIManager* que facilita a obtenção dos dados associados ao POI, a organização dos mesmos e o acesso à listas de POIs de acordo com uma demanda específica, por exemplo, a lista de POIs para desenho.

A classe *Geofence* foi implementada como um *container* genérico para organizar de forma hierárquica os POIs da aplicação, então temos *Geofence*<POI> (lê-se: *Geofence* de POI), mas é possível estender seu uso para outras classes se necessário. Cada *Geofence* tem um pai, que é onde ele está contido. Caso o *Geofence* não esteja contido em nenhum outro, seu pai é o *GeofenceWorld*, um *singleton* (Gamma, E., 1995) que contém todo o mundo.

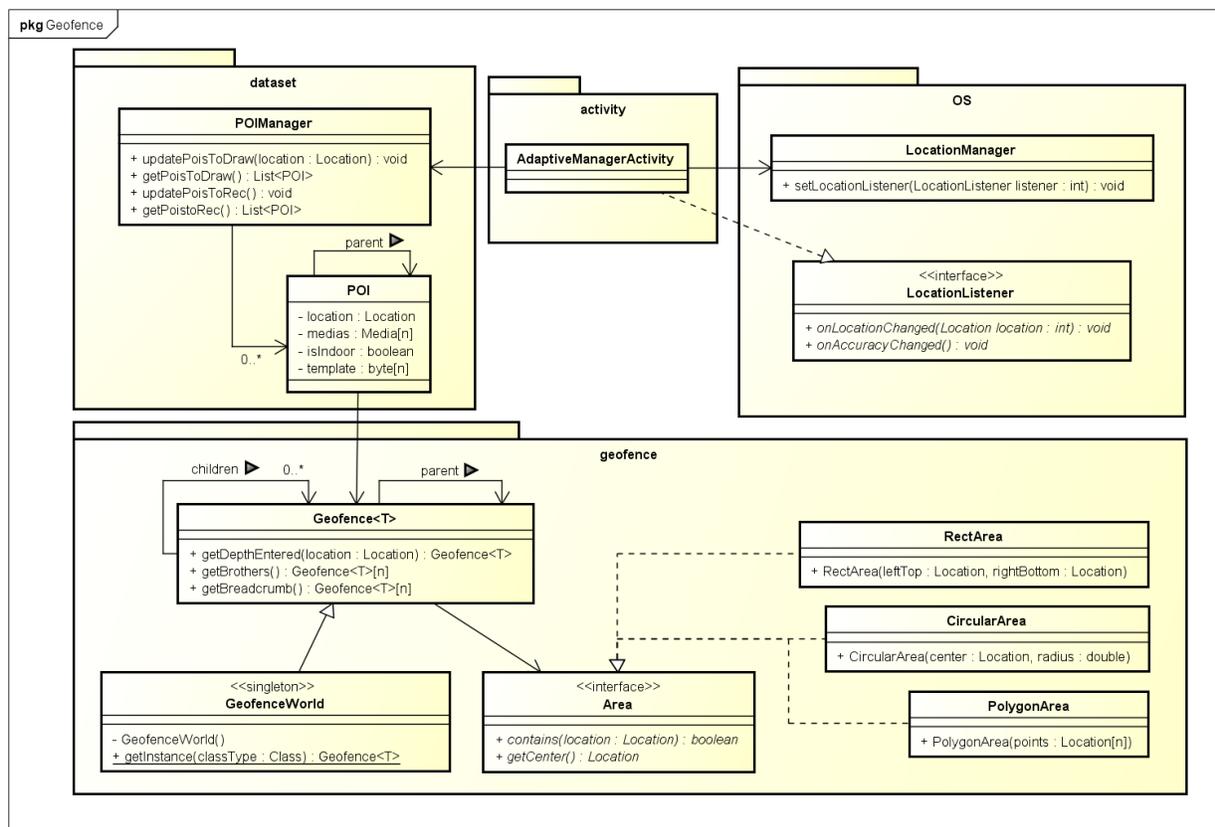


Figura 21. Diagrama de classes da localização do usuário em relação aos *Geofences*.

Com a inclusão de um pai (atributo *trib*), temos por consequência os filhos (atributo *children*), e a classe dá acesso aos irmãos (através do método *getBrothers*), que são os *Geofences* com o mesmo pai do atual. O método recursivo *getDepthEntered* retorna o *Geofence* mais interno que contém uma posição passada por parâmetro. Todo *Geofence* tem uma área que é representada pela interface *Area*. As implementações de *Area* podem ser áreas retangulares, circulares ou poligonais.

A Figura 22 mostra as classes envolvidas no processo de identificação de POIs e objetos associados através do reconhecimento de imagens. A classe *Camera* presente no pacote OS dá acesso aos recursos da câmera do dispositivo e permite que sejam capturados os frames da câmera através da interface *OnCameraPreviewListener*. Toda vez que o frame da câmera está disponível a classe principal passa o mesmo para a classe *POIRecognition* através do método *updateBuffer*.

O Reconhecimento de imagens é feito de forma assíncrona em relação à atualização de frames da câmera, fazendo com que a latência dos métodos de reconhecimento de imagem não influencie na taxa de atualização de frames da sequência de imagens que é mostrada para o usuário.

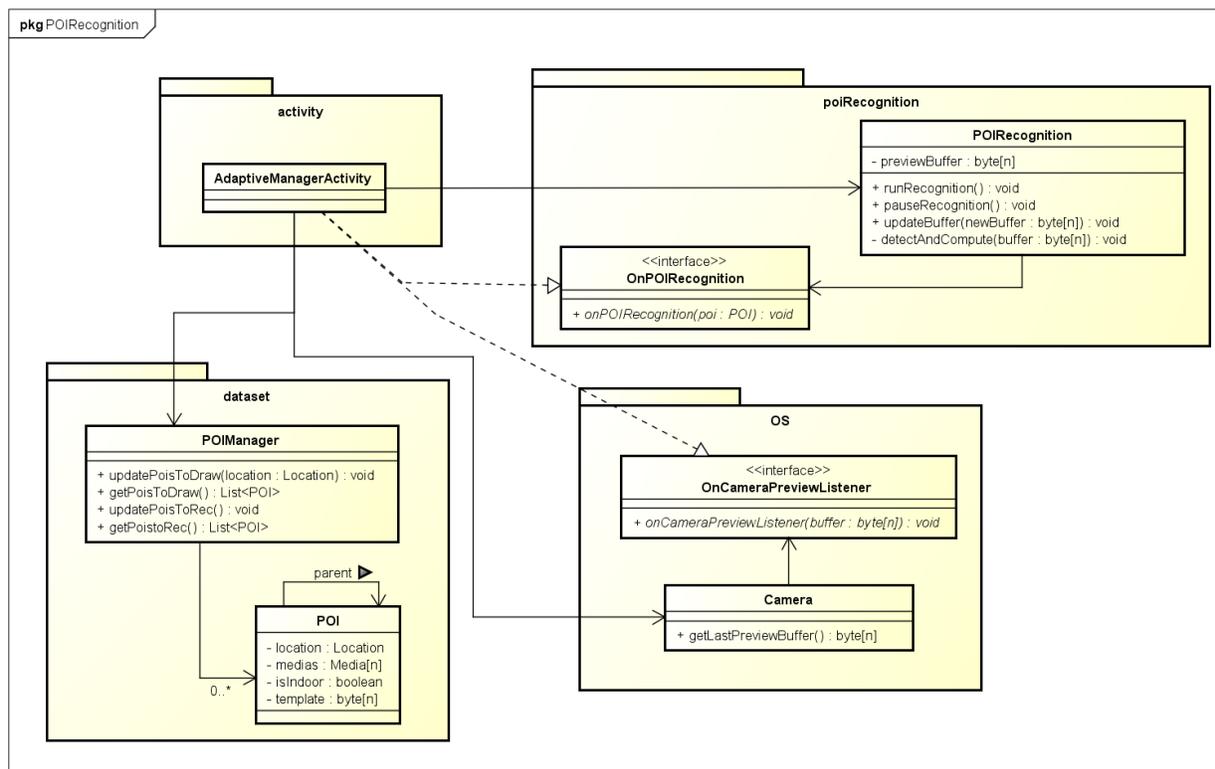


Figura 22. Diagrama de classes do reconhecimento de imagens associadas à POIs.

Os POIs que possuem conteúdo para ser reconhecido e estão a uma distância acessível para tal tarefa são retornados pelo método *getPOIsToRec* da classe *POIManager*. Esses POIs são passados para a classe *POIRecognition* para realizar a tentativa de reconhecimento. Caso ocorra sucesso em no reconhecimento do POI, o POI reconhecido juntamente os detalhes para desenho desse reconhecimento são retornados no método *onPOIRecognition* da interface *OnPOIRecognition*.

A Figura 23 mostra as classes envolvidas na localização interna do usuário. A classe que gerencia localização interna é a *IndoorLocation*, nesta é possível configurar os pontos de acesso, os andares presentes e um objeto responsável pelo recebimento de mudanças na localização interna, o *LocationListener*. Quando a localização interna é ativada (através do método *resume*) é

iniciada uma *thread* (a *ScanThread*) que repetidamente obtém a potência dos sinais dos pontos de acesso.

A lógica da localização interna é controlada através de estados implementados pela classe *LocationState*. Esta classe recebe a potência dos sinais através do método *handleSignals* e dependendo do estado atual realiza a ação adequada. Por exemplo, o estado inicial da localização interna foi definido como “descobrimo andar”, representado pela classe *DiscoveringFloor* que possui a lógica de calcular qual o andar que o usuário se encontra dado as potências do sinais de sensores de curta distância. Quando o andar do usuário é definido a aplicação automaticamente muda de estado para descobrimo a localização, através da classe *DiscoveringLocation*.

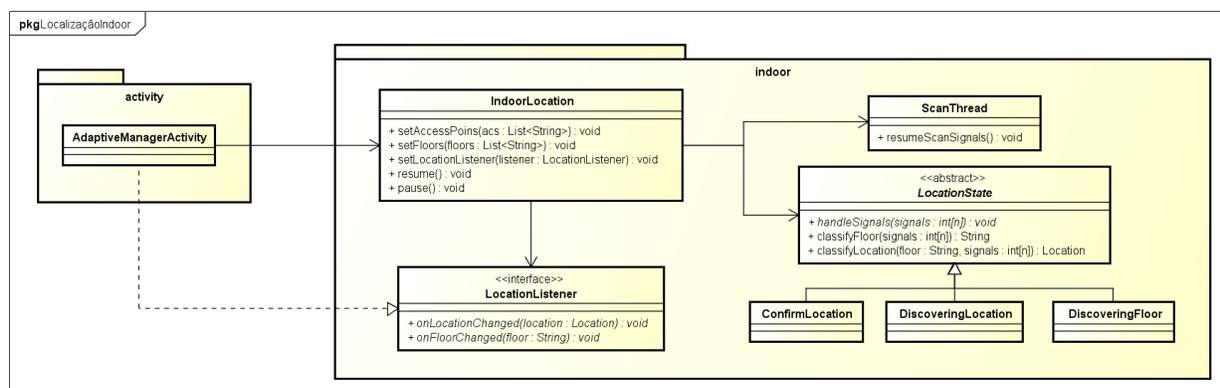


Figura 23. Diagrama de classes do serviço de localização interna.

3.3.3 Diagramas de Sequência

A Figura 24 descreve a visão geral dos diagramas de sequência do modelo. A classe *AdaptiveManagerActivity* realiza as operações de verificação da localização através de sensores de longa distância, atualização da localização e verifica a mudança de ambiente interno e externo. A verificação inicial é para definir por qual meio se deve classificar o ambiente, se o sensor de longa distância falha, possui baixa acurácia ou foi desativado, então uma mensagem é enviada para a *AdaptiveManagerActivity*, para que o reconhecimento possa ser feito por visão computacional. Os passos dessa sequência no caso de erro serão descritos com mais detalhes nos diagramas a seguir. No caso do sensor de longa distância estar funcionando corretamente, o modelo verifica constantemente a mudança de localização.

A partir da localização do usuário são verificados quais *Geofences* (implementação da delimitação da área geográfica de um POI ou AOI) contêm a posição do usuário,

consecutivamente é feita adaptação ao ambiente interno ou externo se for necessário. Sempre que houver uma alteração na posição do usuário em relação à hierarquia de *Geofences* os POIs e AOIs (que agregam POIs internos) são reestruturados para serem desenhados na tela. Caso o usuário esteja dentro de um POI e o mesmo possui localização interna a sequência *Localização Indoor* é acionada. Essas sequências que utilizam a localização serão descritas de forma detalhada em frente.

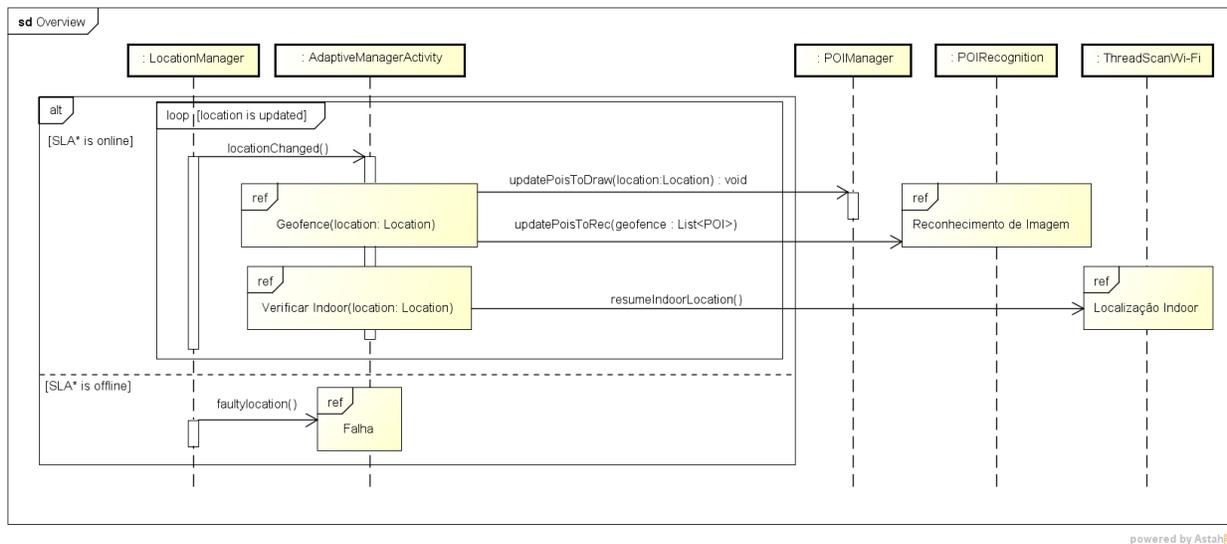


Figura 24. Visão geral dos diagramas de sequência. * SLA (Sensor de Longo Alcance)

A Figura 25 mostra o diagrama de sequência que ilustra como a aplicação se adapta à medida que o usuário transita entre AOIs e POIs. O *LocationManager* notifica o *AdaptiveManagerActivity* sempre que a localização do usuário (obtida por sensores de longo alcance) é alterada. Dessa forma, é solicitada a última posição conhecida e os POI de desenho são atualizados de acordo com essa localização. Para saber quais os POIs devem ser desenhados é feita uma chamada recursiva e hierárquica desde o AOI mais externo verificando se o usuário está dentro dessa AOI, retornando assim o AOI ou POI mais interno que o usuário está contido. A partir desse AOI ou POI são disponibilizados para desenho seus filhos e irmãos na hierarquia.

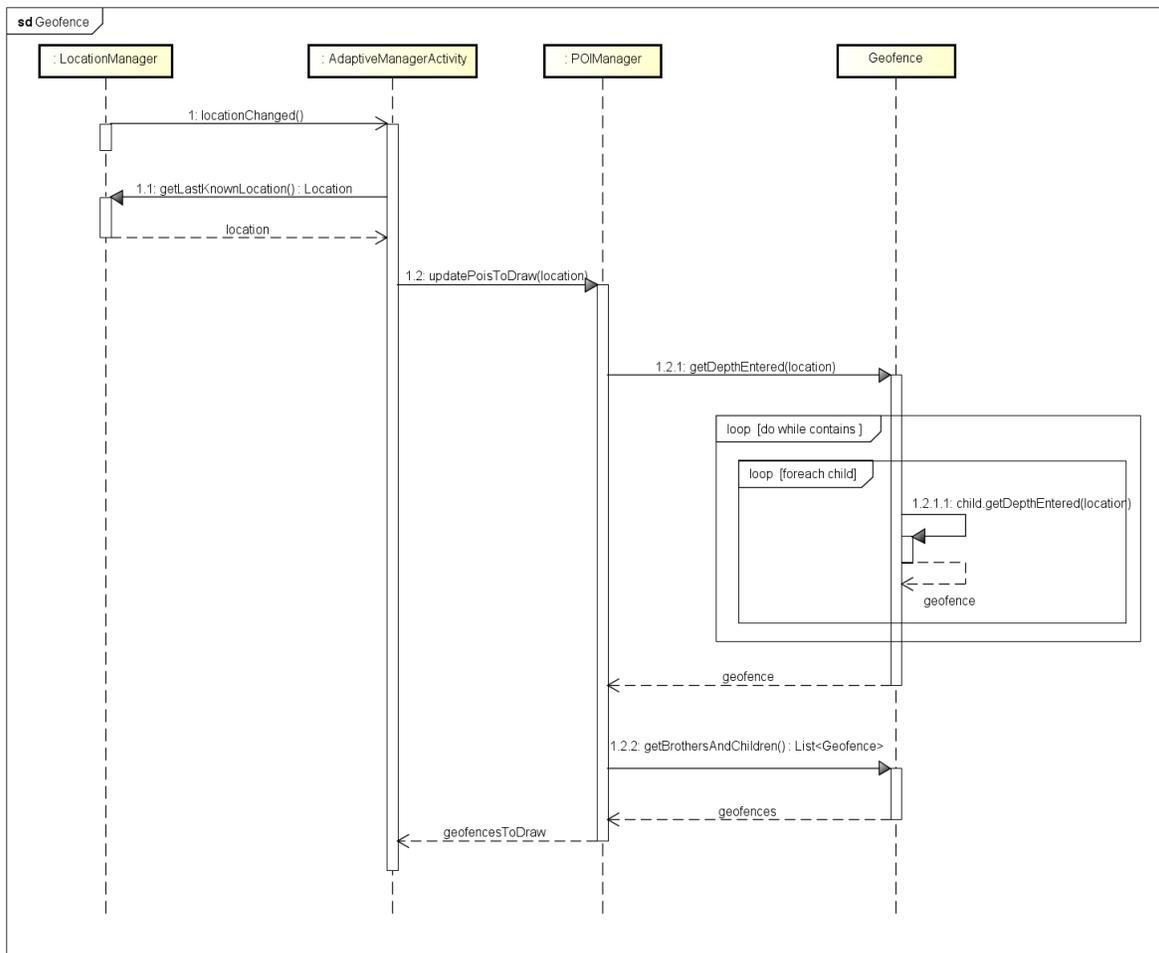


Figura 25. Diagrama de Sequência da lógica de localização do usuário em relação aos *Geofences* dos POIs e AOIs.

Dado que o usuário entrou em um POI que disponibiliza localização interna, é iniciado o serviço de localização que pode ser feito através de sensores de curto alcance, como Wi-Fi, Bluetooth, RF-ID e outros, conforme a Figura 26.

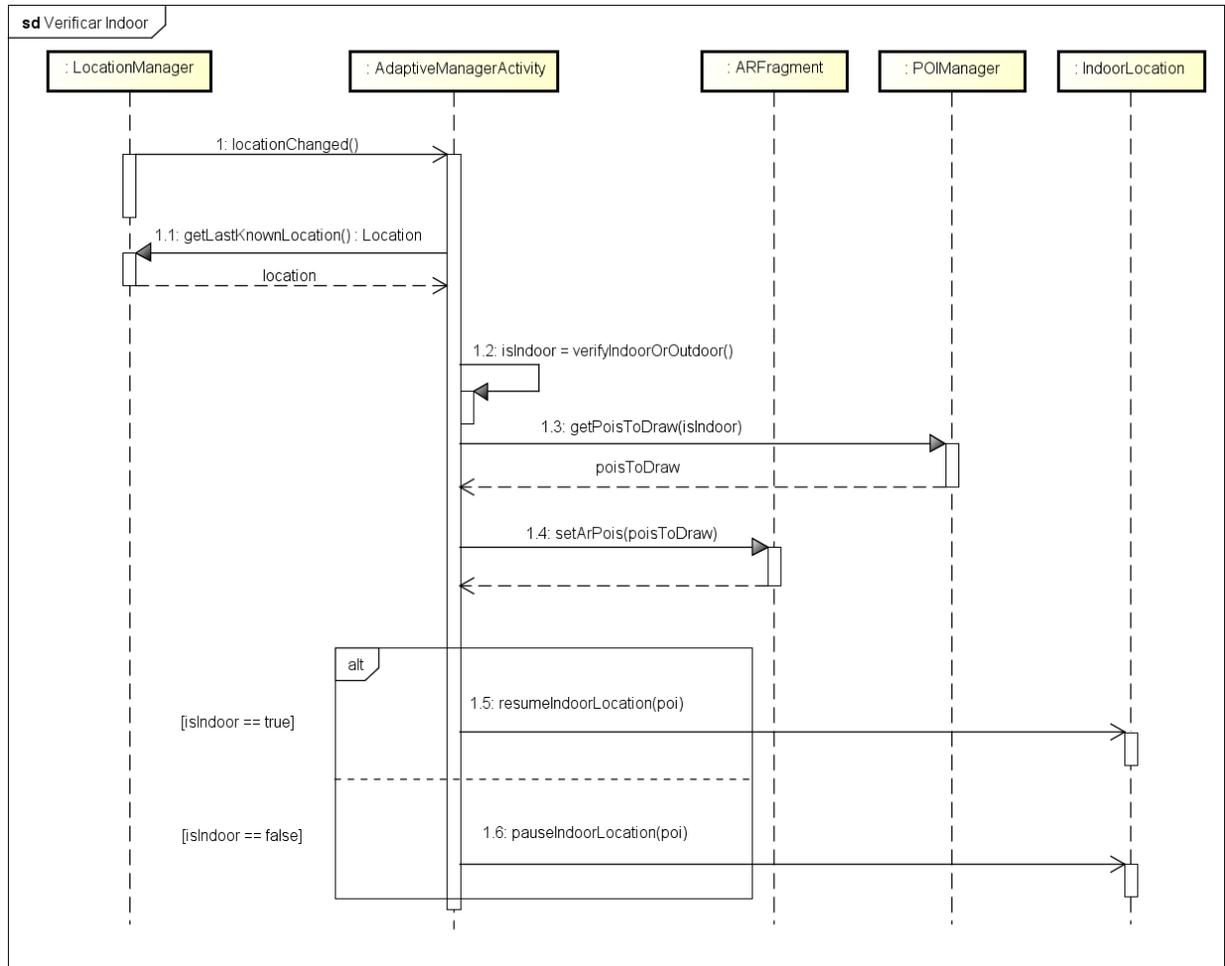


Figura 26. Diagrama de seqüências da lógica que verifica se há disponibilidade de localização interna no POI atual e inicia o serviço em caso positivo.

A Figura 27 mostra o diagrama de seqüência da lógica do serviço de localização em ambientes interiores. O serviço executa em uma *thread* diferente da principal e tem o objetivo coletar os sinais dos pontos de acesso e disponibilizar a para aplicação realizar a classificação da localização.

Primeiramente a aplicação determina o andar atual do usuário e em seguida identifica a localização dentro do POI. Vale ressaltar que essa ordem de verificação pode ser alterada de acordo com a estrutura física do prédio (quantidade de andares e extensão do local).

Após a classificação do andar e local dentro do POI a aplicação pode determinar se o usuário está em uma sala ou corredor e pode disponibilizar conteúdos de acordo com essa posição, assim como, guiar o usuário até um determinado local.

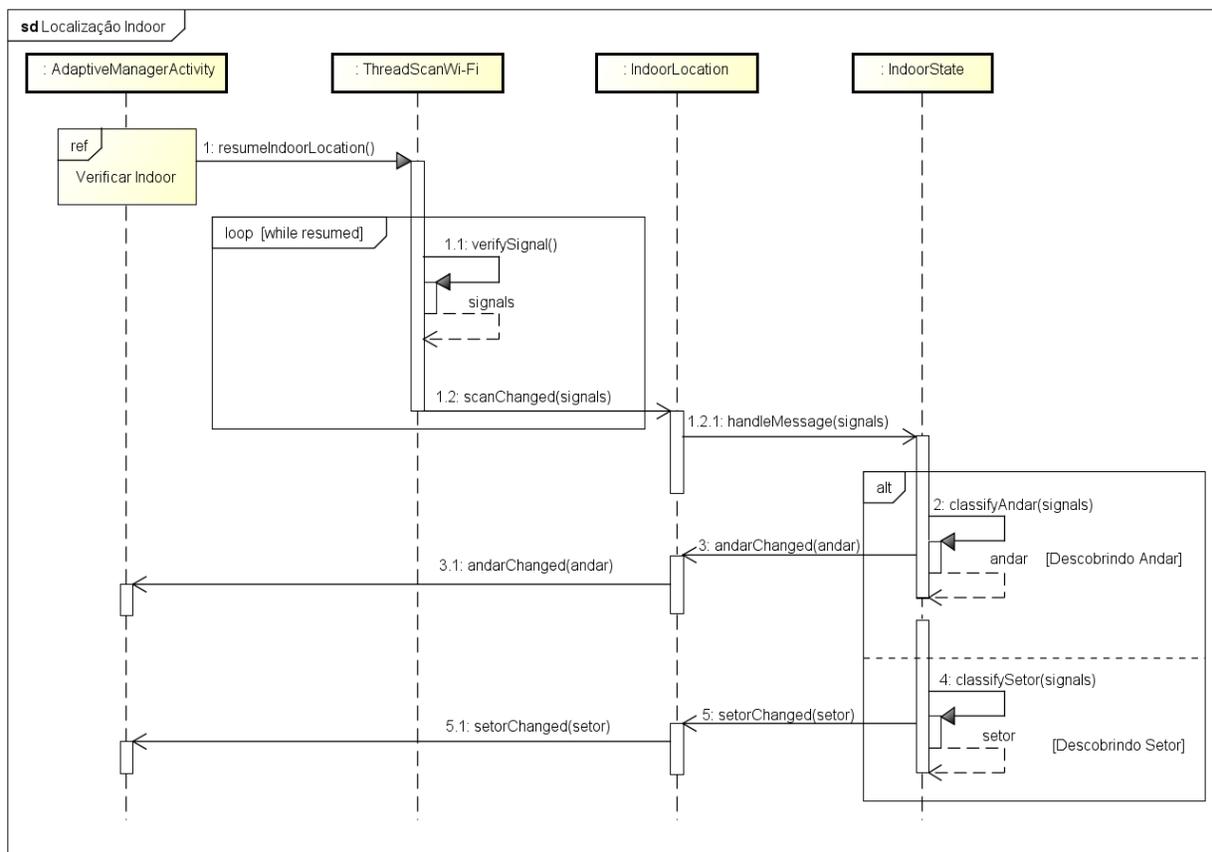


Figura 27. Diagrama de sequência do serviço de localização interna.

Uma vez que a aplicação se adapta a diferentes localizações, seja interno ou externo, o serviço de reconhecimento de imagem pode ser feito de acordo com a disponibilidade de modelos de referência para viabilizar o reconhecimento, conforme explicado na Seção 2.2.1.

A Figura 28 ilustra o diagrama de sequências da lógica do reconhecimento de imagens uma vez que se obteve o AOI ou POI mais interno que o usuário se encontra. Cada AOI ou POI pode conter objetos a serem reconhecidos, assim, para cada AOI ou POI que possui objetos a serem reconhecidos é executado o reconhecimento com a imagem que vem da câmera do dispositivo. Ao realizar o reconhecimento cada matriz de homografia é associada ao seu respectivo POI, permitindo que o processo de síntese da imagem do marcador virtual seja desenhado com a mesma perspectiva que o POI tem ao ser visto pela câmera.

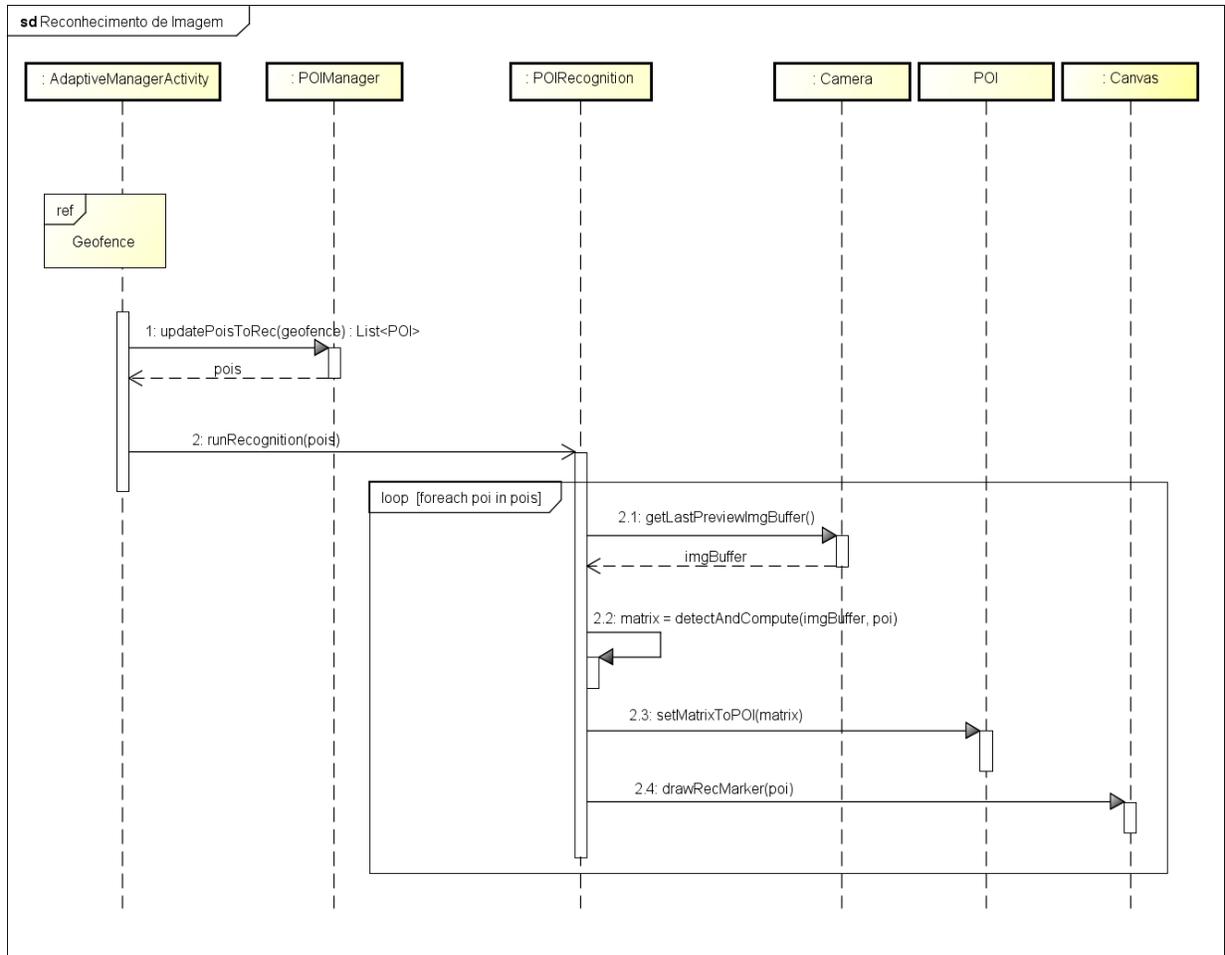


Figura 28. Diagrama de sequência do serviço de reconhecimento de imagem.

A lógica de adaptação depende da localização do usuário em AOIs ou POIs trocando de forma automática a forma de obter o posicionamento do usuário. Caso ocorram falhas na comunicação com sensores de localização (como o GPS) a aplicação pode usar alternativas para saber se o usuário transitou entre ambientes diferentes, como mostrado na Figura 29.

Uma alternativa é utilizar o reconhecimento de imagens para saber se o usuário está em um ambiente interno ou externo. Assim, dado que o usuário estava em um ambiente interno e ocorreu uma falha nos sensores de posicionamento, o algoritmo de classificação de cena obtém alguns frames da câmera do usuário para classificar entre ambiente interno e externo. Se o classificador concluir que o usuário está em um ambiente interno, então a lógica da aplicação não é alterada, mas se for um ambiente externo então a aplicação deve se adaptar, considerando que o usuário saiu do POI que estava, por exemplo.

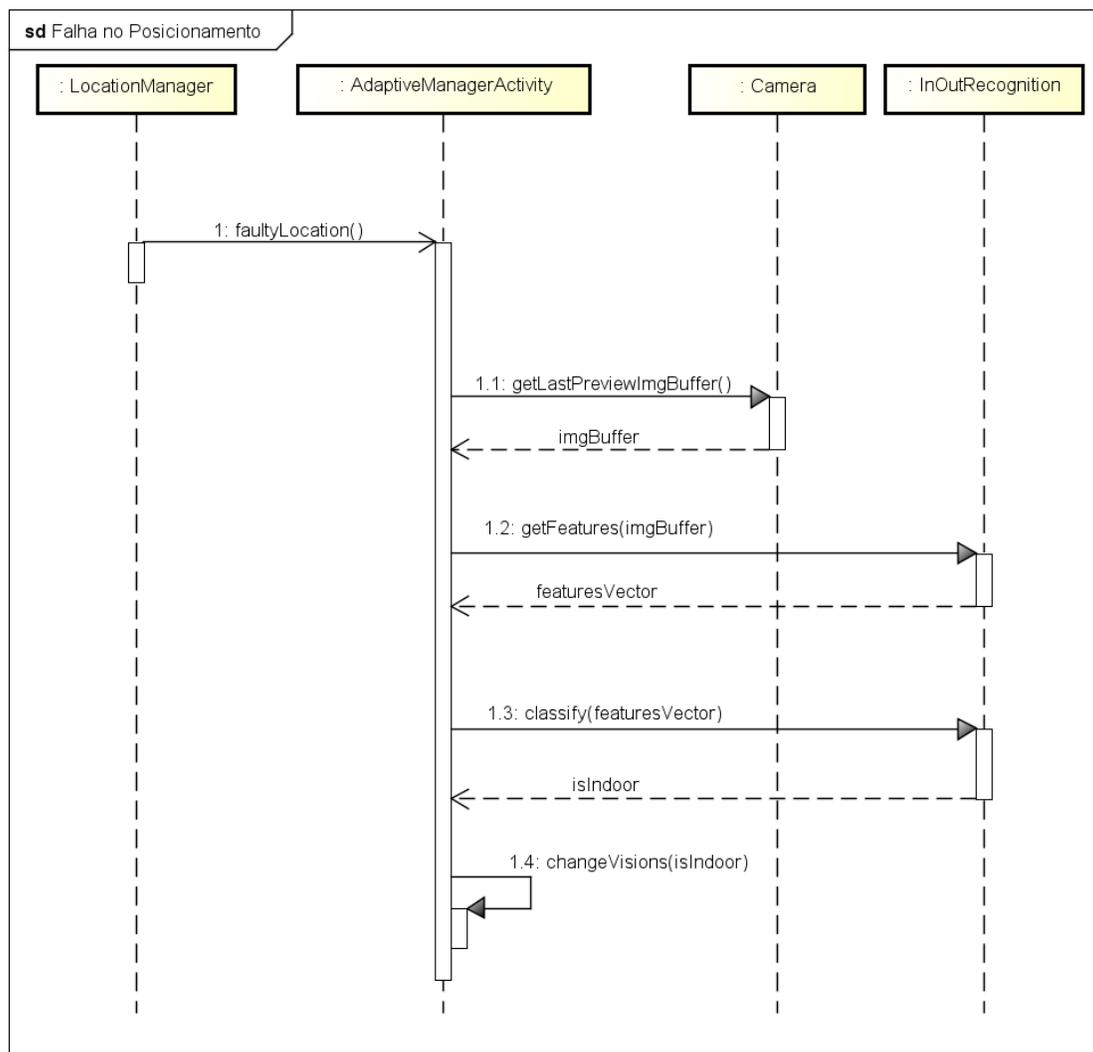


Figura 29. Diagrama de Sequência de Erro no GPS

4. PROTÓTIPO DE VALIDAÇÃO

Com a finalidade de testar e validar o modelo proposto foi desenvolvida uma aplicação de RAM que utiliza diversas tecnologias disponíveis em dispositivos móveis, tais como: sensores inerciais, câmera, GPS, Wi-Fi, acesso à Internet, entre outros. A aplicação também faz uso de algoritmos de visão computacional para reconhecimento de padrões em imagens, principalmente quando há baixa precisão nos sensores ou falha em obter dados dos mesmos.

De maneira geral, será avaliada a capacidade da aplicação se auto ajustar aos diferentes cenários de ambiente externo, interno, à transição entre eles e às condições de distância do usuário em relação a um determinado POI. Ou seja, caso o usuário esteja em um ambiente externo, a aplicação deve mostrar informações sobre os POIs e realizar o registro dos elementos virtuais de forma adequada, por exemplo, utilizando GPS e sensores inerciais.

Da mesma forma, quando o usuário está dentro de um prédio, a aplicação deve se ajustar para mostrar o conteúdo interno deste prédio realizando o registro de forma adequada, e como dentro de um prédio o GPS geralmente possui baixa acurácia, a estratégia de rastreamento da RAM deve ser alterada para uma estratégia de localização interna ou de visão computacional, melhorando a acurácia da localização.

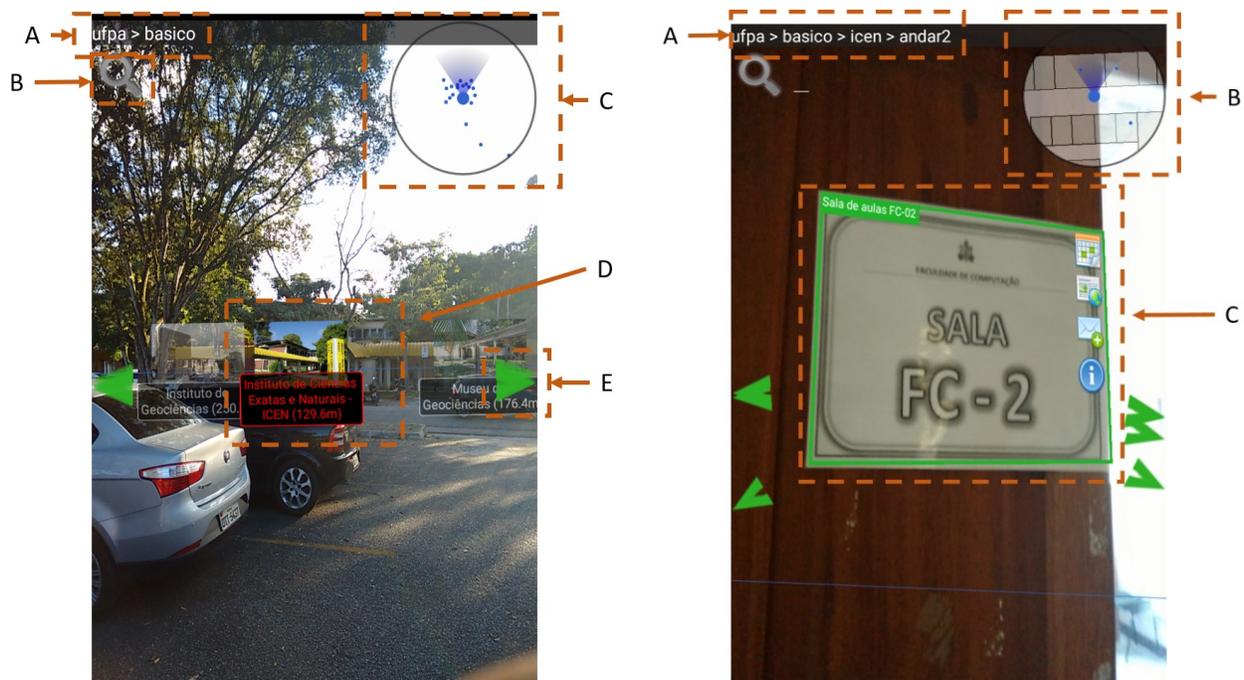
As seções a seguir descrevem as principais funcionalidades da aplicação que foram utilizadas no teste de validação do modelo.

4.1 Interface da Aplicação

A aplicação desenvolvida possui o navegador de RA que se adapta às transições do usuário entre AOIs e POIs e possui funcionalidades de busca textual de POIs, rota até o POI e visualização do conteúdo de um POI.

A Figura 30 (a) mostra os principais elementos gráficos da aplicação quando em um ambiente externo. Em “A” é mostrada a localização atual do usuário de forma hierárquica, em “B” o botão responsável por iniciar a busca por POIs, em “C” o mini radar que mostra uma visão aérea da posição dos POIs em relação ao campo de visão do usuário, em “D” o marcador virtual de um POI que está marcada de vermelho para indicar que este é o próximo passo da rota desejada e em “E” é mostrada as setas que indicam que existem mais POIs na direção que a seta está apontando.

A Figura 30 (b) mostra as mudanças que ocorrem na aplicação quando o usuário transita para um ambiente interno. Em “A” a localização está mais específica, situando o usuário no segundo andar de um prédio. Em “B” o radar foi adaptado para mostrar a planta baixa do prédio. Em “C” é mostrado o marcador virtual adaptado ao reconhecimento de imagens, que ocorre quando o usuário aponta sua câmera para algum objeto que pode ser reconhecido e está associado a um POI ou IOI.



(a)

(b)

Figura 30. Interface gráfica da aplicação protótipo. (a) mostra a aplicação funcionando em um ambiente externo e seguindo uma rota, (b) mostra a aplicação funcionando em ambiente interno e reconhecendo uma placa associada a um POI.

A Figura 31 (a) mostra a lista de POI encontrados pela busca quando o usuário insere um termo de busca, neste caso o usuário inseriu a palavra “instituto”. A Figura 31 (b) mostra a tela

responsável por apresentar o conteúdo de um POI para o usuário, neste o usuário pode acessar um texto descritivo, imagens do POI e pode traçar uma rota até este POI.



Figura 31. Em (a) é apresentada a lista de POIs que foram encontrados com a busca para palavra “instituto” e em (b) é apresentado o conteúdo do POI “Reitoria” onde é possível ler um texto sobre o POI, ver imagens e traçar uma rota até o POI.

4.2 Organização Hierárquica do Cenário de Teste

O protótipo foi validado com testes dentro da UFPA (Universidade Federal do Pará) campus Guamá, dessa forma, os AOIs e POIs foram dispostos de acordo com a organização hierárquica comumente conhecida nesta instituição de ensino. A Figura 32 mostra a estrutura hierárquica que foi utilizada pelo protótipo de validação, no primeiro nível está a UFPA, seguido pelos seus setores (básico, profissional, saúde e tecnológico) e pelos POIs dentro da universidade.

Os testes que envolvem localização interna foram testados no ICEN (Instituto de Ciências Exatas e Naturais) que se localiza no setor básico da UFPA. O ICEN possui dois andares e suas antenas de Wi-Fi se localizam apenas no segundo andar. Além disso, em uma das tarefas os participantes dos testes foram conduzidos à sala de aulas FC-2 que fica no segundo andar, ao

LABVIS (Laboratório de Visualização, Interação e Sistemas inteligentes) que fica no segundo andar e ao laboratório da pós-graduação em ciência da computação que fica no primeiro andar.

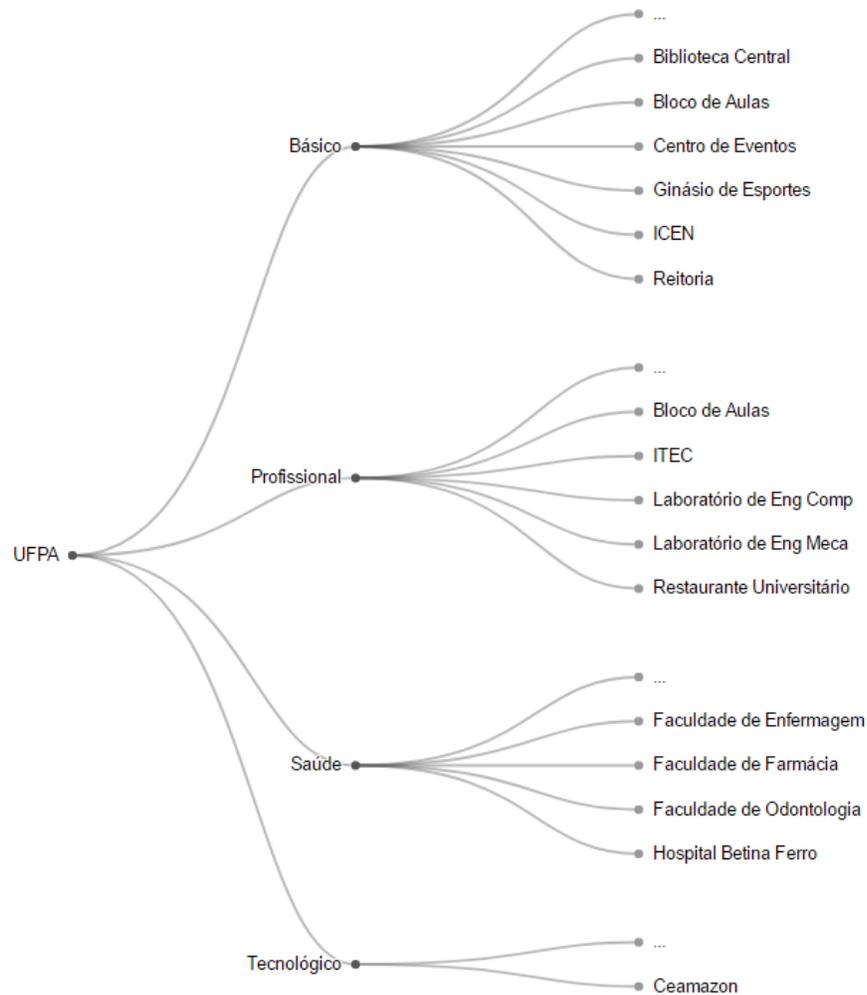


Figura 32. Organização hierárquica dos POIs utilizados no protótipo de validação.

4.3 Navegação com RA

Essa funcionalidade auxilia o usuário a ter consciência da existência de POIs e guia o usuário até os POIs. A funcionalidade de navegação contempla ações nos ambientes externos, internos, e as mudanças de um ambiente para outro.

Na aplicação, o usuário pode visualizar a posição e direção de AOIs e POIs com a movimentação realizada no dispositivo, isso acontece porque a sequência de imagens capturadas pela câmera é misturado com marcadores virtuais posicionados na direção real dos AOIs e POIs,

possibilitando ao usuário encontrar os mesmos com base na localização atual e na direção da câmera do dispositivo móvel.

A Figura 33 apresenta quatro capturas de tela da aplicação na funcionalidade de navegação de RA para diferentes cenários, sendo que a aplicação se adaptou a tais cenários sem a necessidade do usuário intervir ou fornecer informações. Neste caso, o usuário está indo à sala FC-2 do Instituto de Ciências Exatas e Naturais (ICEN) que fica no campus básico e está vindo do campus profissional.

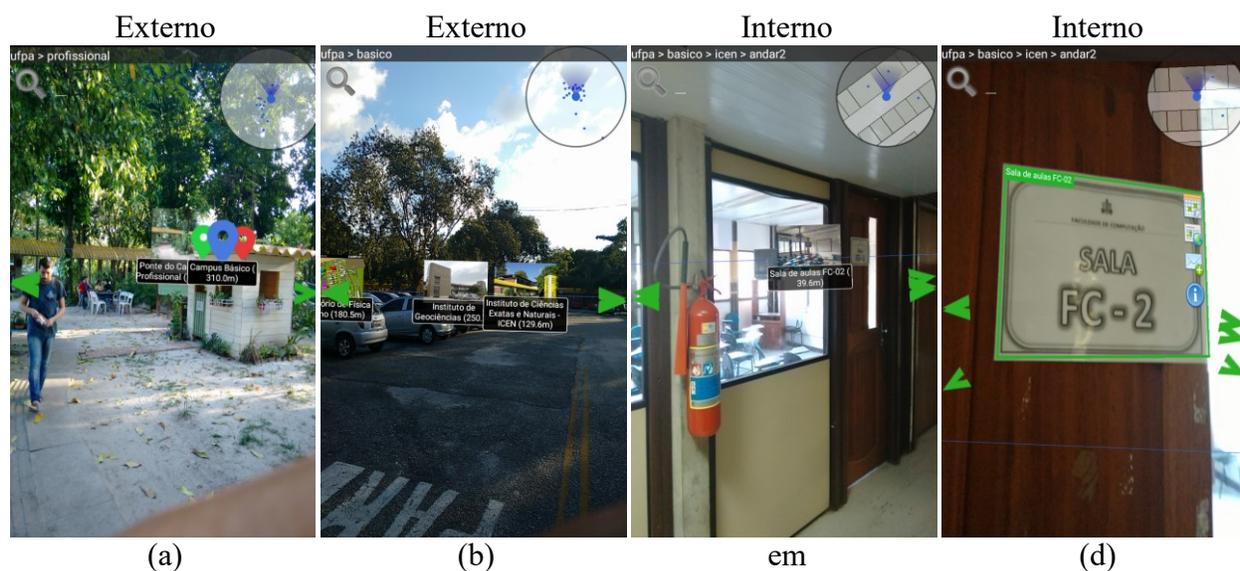


Figura 33. Capturas de tela do protótipo de validação no setor profissional apontando para o setor básico (a), no setor básico apontando para o prédio do ICEN (b), dentro do ICEN, apontando para a sala FC-2 (c) e reconhecendo a placa da sala (d).

Na Figura 33 (a) o usuário está na AOI do setor profissional, assim o POI ICEN é representado pelo marcador da AOI que o contém, neste caso o AOI do setor básico. Quando o usuário entra na AOI do básico (b), os POIs dentro desta são revelados, inclusive o marcador do ICEN. Assim que o usuário entra no ICEN (c), a localização interna é ativada, permitindo que os IOIs sejam posicionados em relação ao usuário. Dessa forma, o usuário é guiado até a sala FC-2 e pode realizar o reconhecimento de imagens somente apontando a sua câmera para alguns pontos de referência pré-cadastrados (d).

Durante qualquer momento dessa trajetória o usuário pode obter informações sobre o POI clicando no respectivo marcador virtual. Observa-se que dependendo do local do usuário em relação aos AOIs e POIs diferentes marcadores são apresentados, consequentemente o detalhamento do conteúdo também é apresentado conforme essa localização do usuário, por

exemplo, quando o usuário está fora do prédio são mostradas informações sobre os locais existentes, eventos, horário de funcionamento, entre outros. Mas quando o usuário está de frente para a sala, informações mais específicas sobre a sala são apresentadas para o usuário.

Quando o usuário deseja mais informações sobre um POI, caso o mesmo esteja visível na tela do dispositivo móvel, o usuário pode selecionar mesmo através de um toque no ícone virtual do referido POI. Ao selecionar um POI o usuário pode ter acesso ao conteúdo multimídia.

4.4 Lógica Adaptativa

O Algoritmo 1 mostra o pseudocódigo responsável pela adaptação da aplicação entre ambientes internos e externos com a utilização das atualizações da posição medida por GPS e o acionamento automático da localização interna quando o usuário está dentro de POI que possui suporte para tal. Caso a localização de GPS não esteja com boa acurácia (por exemplo, acurácia maior que 20 metros) e a localização interna não esteja funcionando, o reconhecimento de cena é feito para identificar se o usuário mudou de ambiente. Ao final a função *changeVisions()* é chamada para adaptar os componentes de interface gráfica da aplicação de acordo com o ambiente passado por parâmetro.

Algoritmo 1 - Adaptação da Interface Conforme a Localização Interno-Externa.

```
// Gerenciador de Localização do Dispositivo
LocationManager locationManager = System.getLocationManager();
// Fragmento para desenhar os POI na visão de RA
ARFragment arfragment = getFragmentManager.get("ARFragment");
IndoorLocation indoorLocation; // Localizador interno e externo
POIRecognition poiRecognition; // Reconhecedor de objetos em imagens
Camera istri = System.getCameraInstance(); // Câmera do dispositivo
// Classificador de imagens para ambiente interno e externo
InOutRecognition inOutRecognition;

boolean indoorLocationResumed;

// A configuração de um istribuí é necessária para verificar
// as mudanças de localização e agir de acordo
// O istribuí é executado toda vez que a localização do usuário é alterada.
locationManager.setOnLocationChanged( istri {

    boolean isIndoor; // Flag para posição interna ou externa

    if ( locationManager.hasGoodAccuracy() && !indoorLocationResumed) {

        // Detecção do Geofence do usuário a partir da localização
        Location location = locationManager.getLastKnownLocation();
        List<Geofence> geofencesToDraw =
```

```

POIManager.updatePOIsToDraw(location);

// Reconhecimento objetos dos POIs do Geofence
List<POI> poisToRec = POIManager.updatePOIsToRec(geofencesToDraw);
poiRecognition.runRecognition(poiToRec);
// Verifica se a posição atual corresponde a um ambiente
// interno ou externo
isIndoor = verifyIndoorOutdoor(location);
POI currentPOI = POIManager.getCurrentPOI(); // POI atual

if (isIndoor) {
    // Ativar localização Indoor
    indoorLocation.resumeIndoorLocation( callback {

        // POIs para desenhar na visão de RA
        List<POI> poisToDraw = POIManager.getPOIsToDraw(isIndoor);
        arfragment.setPOIstoDraw(poisToDraw);
        indoorLocationResumed = true;
    });
} else {
    // Pausa a localização interna
    indoorLocation.pauseIndoorLocation(currentPOI);
}
} else if (!indoorLocationResumed) {
    // Acionado no caso de GPS falhar, é realizada
    //classificação de ambiente por reconhecimento de imagem
    byte[] imgBuffer = camera.getLastPreviewImgBuffer();
    float[] featuresVector = inOutRecognition.getFeatures(imgBuffer);
    isIndoor = inOutRecognition.classify(featuresVector);
}
// Mudança de visão no caso de mudança de ambiente
changeVisions(isIndoor);
});

```

4.5 Localização Interna

Para obter uma boa acurácia de localização em ambientes internos foi implementada na aplicação uma técnica de localização interna e uma lógica de transição entre a localização externa e interna, tendo em vista que a localização externa já possui boa acurácia para aplicações de RAM, enfrentando problemas somente quando se está muito próximo aos POIs.

Dentre as diversas estratégias de localização interna existentes na literatura, foi utilizado no protótipo de validação do modelo o método de *fingerprnt* com sinais de Wi-Fi (VILLARRUBIA et al., 2014) (CHEN et al., 2014), uma vez que geralmente os locais internos já possuem uma infraestrutura de rede local que fornece Internet e pode ser utilizado apenas a potência do sinal de cada AP (Ponto de Acesso, do inglês *Access Point*) sem modificar a estrutura existente.

A localização interna pode ser utilizada em qualquer prédio que contenha Aps espalhados que emitam sinais de curta distância (como Wi-Fi), sendo que quanto maior o número de Aps e quanto mais espalhados pelo prédio eles estão, maior será a acurácia e precisão da localização. Para conduzir os testes no protótipo de validação, foi implementada a localização interna no prédio do ICEN.

O ambiente foi mapeado com a coleta da intensidade dos sinais dos Aps em locais predefinidos (como mostra a Figura 34). Foram coletadas 200 amostras de sinais de cada ponto demarcado na Figura 34. O vetor de entrada para o algoritmo de aprendizado de máquina possui 55 dimensões referentes aos AP presentes no ICEN, e o valor dessas dimensões varia entre -120 (valor atribuído mesmo quando não há sinal do AP) e -1 que é o sinal de maior intensidade medido em decibéis.

Assim, foi construída uma base de dados com um conjunto de sinais e a localização do usuário. Essa base foi utilizada para treinar alguns algoritmos (foi selecionado o de melhor acurácia) de aprendizado de máquina e gerar um classificador, de forma que dado novas potências dos sinais dos Aps para o modelo, este infere o local onde o usuário está. O algoritmo também é robusto a mais de um andar no prédio.

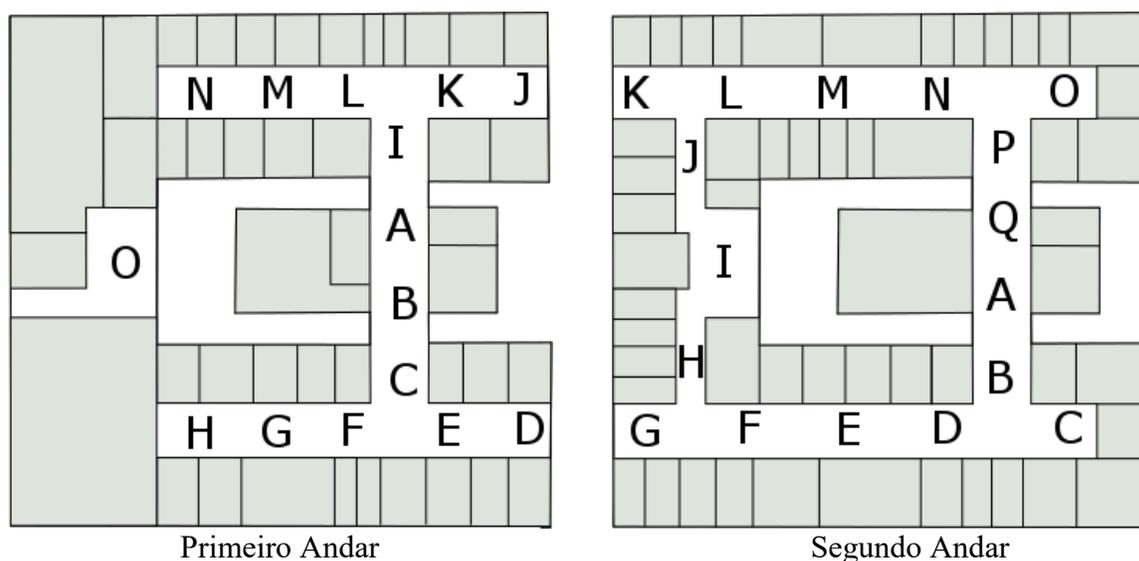


Figura 34. Planta dos dois andares do ICEN com as localizações discretas (letras maiúsculas) utilizadas como treinamento e saída dos classificadores da posição interna. Para cada localização foram registradas as potências do sinal dos Aps de Wi-Fi para serem usados no treinamento do classificador.

Para obter um classificador com boa acurácia e precisão, o software Weka (Hall et al., 2009) foi utilizado para realizar o treinamento e avaliação dos modelos treinados. Para verificar se a

classificação é possível mesmo sem uma seleção de modelo (*model selection*) mais apurada, foram testados alguns modelos que não possuem muito parâmetros para serem configurados, são eles: KNN (*K Nearest Neighbors*) (Larose, 2014), *Random Forest* (Breiman, 2001), *Árvore de Decisão* (Russel e Norvig, 2009) e *Naive Bayes* (Rish, 2001).

Para classificar tanto a localização e o andar, foram treinados três modelos de classificação, sendo um para classificar qual o andar que o usuário está, e dois para classificar a localização dado qual o andar anteriormente classificado, uma vez que o prédio do ICEN tem dois andares.

Visando treinar o classificador de andar, foram utilizadas todas as coletas realizadas nos dois andares do prédio, sendo que neste caso os rótulos não identificavam o local, somente o andar. Para treinar cada classificador de localização em um dado andar, foram fornecidos somente os sinais coletados no andar específico e os rótulos representavam o local onde o mesmo foi coletado.

A Tabela 2 mostra as métricas obtidas pelo software Weka quando realizada uma validação cruzada de 10 *folds*. Na coluna “Parâmetros” são apresentados os parâmetros utilizados no treinamento (Foram mantidos os valores padrão do Weka). Em todos os três casos o classificador *Random Forest* obteve um melhor desempenho, mesmo quando o número de árvores aleatórias é baixo (20), portanto esse classificador foi utilizado na implementação do protótipo de validação.

Tabela 2. Métricas obtidas pelo treinamento dos classificadores para obter a localização interna.

Tipo de Classificação	Classificador	Parâmetros	Precisão	Revocação	F-Measure
Andares	<i>Random Forest</i>	I=20	0.999	0.999	0.999
	Árvore de Decisão	C=0.25	0.996	0.996	0.996
	KNN	K=3	0.987	0.987	0.987
	<i>Naive Bayes</i>	-	0.738	0.688	0.680
Primeiro Andar	<i>Random Forest</i>	I=20	0.999	0.999	0.999
	Árvore de Decisão	C=0.25	0.997	0.997	0.997
	KNN	K=3	0.981	0.981	0.981
	<i>Naive Bayes</i>	-	0.917	0.900	0.899
Segundo Andar	<i>Random Forest</i>	I=20	0.999	0.999	0.999
	Árvore de Decisão	C=0.25	0.996	0.996	0.996
	KNN	K=3	0.990	0.990	0.990
	<i>Naive Bayes</i>	-	0.961	0.961	0.960

Após finalizar o treinamento do modelo, o mesmo foi exportado pelo Weka para ser lido dentro da aplicação que funciona no sistema Android. Para validar a acurácia dos classificadores na prática, foram realizadas 186 medições usando a aplicação para verificar se a acurácia se mantém na prática. Com essas medições obteve-se uma taxa de acerto de 83,33%.

4.6 Algoritmos de Registro baseado em Visão

O *pipeline* do reconhecimento de imagens possui etapas de detecção de características, descrição de características, correspondência de características e de cálculo da posição da câmera.

4.6.1 Detecção e Descrição de Características

A detecção de características consiste em localizar pontos distinguíveis em uma imagem, de tal forma que dado outra imagem da mesma cena, porém com outro ângulo de captura, seja fácil (usando comparações simples) identificar onde estão os mesmos pontos previamente identificados. A Figura 35 mostra uma imagem e suas características detectadas.



Figura 35. Imagem em escala de cinza e suas características detectadas e representadas por círculos coloridos.

A descrição de características consiste em extrair informações que representem os pontos detectados, possibilitando a comparação entre os mesmos para saber quais são mais semelhantes ou divergentes. Geralmente a descrição de características gera um vetor de valores que podem ser binários ou reais dependendo da abordagem utilizada. Então, dado o exemplo da Figura 35 descrever as características significa extrair um vetor de valores de cada ponto colorido na

imagem. Esse vetor de valores é extraído com base na intensidade dos pixels ao redor do pixel central de cada ponto.

Atualmente existe uma quantidade expressiva de algoritmos que realizam estas tarefas (Krig, 2014). Entretanto, essa gama de opções gera dificuldades na escolha do melhor algoritmo. Assim, esta tese realiza o teste com três desses algoritmos que têm implementações prontas na biblioteca de visão computacional OpenCV (Bradski, 2000). São eles ORB (Rublee et al., 2011), AKAZE (Alcantarilla, Nuevo, Bartoli, 2013) e o BRISK (Leutenegger, Chli, Siegwart, 2011). Os três algoritmos utilizam descritores binários que são mais rápidos e utilizam menos memória, sendo mais apropriados para dispositivos móveis que têm baixo recurso computacional. Dessa forma, os três algoritmos foram escolhidos por questões de velocidade de execução e facilidade de implementação.

O ORB é uma combinação de duas outras técnicas: o FAST (Rosten, Porter, Drummond, 2010) (um detector rápido de características) e o BRIEF (Calonder et al., 2012) que descreve as características de pixels na imagem através de testes binários simples. Além disso, o ORB realiza um teste para tornar o BRIEF invariante à rotação. Segundo as comparações feitas por Rublee et al. (2011) o ORB é duas vezes mais rápido que o SIFT e é uma ordem de magnitude mais rápido que o SURF.

O BRISK é outra técnica de detecção e descrição de pontos de interesse em imagens baseado no FAST e no BRIEF. O algoritmo usa amostragem binária feita à mão para a descrição das características. Essa descrição vai ficando mais densa e especializada quanto mais próximo da característica detectada. Este também é uma ordem de magnitude mais rápido que o SURF e surgiu da mesma necessidade de reduzir o tempo computacional do SIFT e do SURF, porém tentando manter o mesmo nível de precisão.

O AKAZE, como os anteriormente citados, possui computação mais rápida do que os conhecidos SURF e SIFT e também é mais rápido que seu predecessor KAZE, além de ser mais robusto que o ORB e o BRISK para mudanças de perspectiva e ruídos na imagem. Utilizando o FED (*Fast Explicit Diffusion*) consegue melhorar a detecção de características num espaço de escala não linear.

4.6.2 Correspondência entre Características

Para realizar o reconhecimento de imagens é necessário saber onde está uma imagem de referência na imagem de cena, para isso são feitas comparações no intuito de encontrar quais características são comuns entre as imagens.

As comparações utilizam os vetores de descrição obtidos na etapa anterior e devem ter um bom desempenho, uma vez que a complexidade de comparações é de $O(NM)$ sendo N o número de características de uma imagem e M o número de características da outra imagem. Por isso, optou-se pelo uso de algoritmos que empregam vetores binários, dado que para comparar vetores binários utiliza-se a distância de Hamming, que faz uso de operações binárias na comparação e é mais eficiente que a distância euclidiana.

A Figura 36 mostra um exemplo de casamento de características sendo aplicado em duas imagens, sendo uma de cena (esquerda) e uma de referência (direita). Nesta foram desenhadas linhas entre as correspondências entre as duas imagens, e observa-se que as correspondências foram bem sucedidas em identificar onde estão os pontos da imagem de referência na imagem da cena.

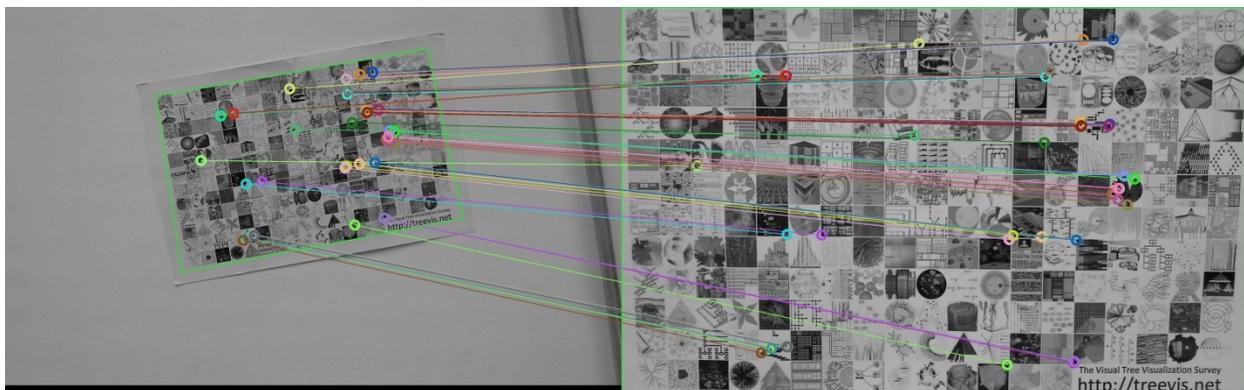


Figura 36. Imagem de cena (esquerda), imagem de referência (direita) e as correspondências de suas características (linhas coloridas).

4.6.3 Transformação Homográfica

A transformação homográfica é um tipo de transformação que permite aplicar perspectiva em objetos. Ela pode ser utilizada para transformar um plano tridimensional que está projetado em duas dimensões, alterando o ângulo de visão. A transformação homográfica em computadores geralmente é calculada a partir de uma matriz H , tal que ao multiplicar a matriz de pontos X de um objeto com a matriz H , tem-se a matriz de pontos do objeto transformado X' .

A matriz H apresenta oito graus de liberdade que representam o número de parâmetros que podem variar de forma independente, dando a esta transformação a possibilidade de realizar transformações em projeções perspectivas.

Em tarefas de síntese de imagem é necessário saber de antemão a matriz H para realizar as transformações desejadas na cena. Em visão computacional deseja-se obter essa matriz dado duas imagens e para isso utilizam-se as correspondências entre as duas imagens para gerar um sistema linear da seguinte forma.

$$H \begin{bmatrix} x_1 & x_2 & \dots \\ y_1 & y_2 & \dots \end{bmatrix} = \begin{bmatrix} x'_1 & x'_2 & \dots \\ y'_1 & y'_2 & \dots \end{bmatrix}$$

Assim, dado que se conhece os valores dos pontos $\{(x_1, y_1), (x_2, y_2), \dots\}$ de uma imagem e os pontos correspondentes $\{(x'_1, y'_1), (x'_2, y'_2), \dots\}$ de outra imagem, é possível obter a transformação H resolvendo o sistema linear. Essa correspondência entre pontos pode ser vista na Figura 36.

Atualmente a biblioteca OpenCV possui dois métodos robustos para calcular a matriz homográfica entre uma imagem de referência e sua correspondência na imagem da cena. Os métodos são o RANSAC (Choi, 1981) e o Least Medians (LMEDS) (Rousseeuw, 1984).

O RANSAC (Choi, 1981) é um método iterativo que pode estimar parâmetros. Ele seleciona amostras que são classificadas como *inlier* e *outlier* e calcula a homografia. Ele necessita de dois parâmetros a serem configurados: o limiar para classificar as amostras entre *outliers* e *inliers* e o número de iterações do algoritmo. Caso o número de iterações seja baixo a solução final tem menos chances de ser a solução ótima, contudo é mais eficiente em tempo.

O LMEDS (Rousseeuw, 1984) é uma abordagem que surgiu a partir da tentativa de melhorar a técnica clássica já existente de mínimos quadrados, que visa minimizar os valores dos erros. Ela funciona muito bem quando há menos de 50% de *outliers* (falsas correspondências) encontrados. Uma vantagem do LMEDS sobre o RANSAC é que este não precisa estimar um limiar ou determinar o número de iterações, a desvantagem é que ele não é capaz de lidar com situações em que os *outliers* correspondem a mais da metade dos dados (Dubrofsky, 2009).

4.7 Reconhecimento de Imagens

O reconhecimento de imagens deve ocorrer de forma automática, sem a necessidade de indicar qual objeto deve ser reconhecido e quando o processo deve ser iniciado. Assim, para

obter o modelo utilizado no processo de reconhecimento a aplicação utiliza as informações de localização (AOIs, POIs e IOIs que o usuário está próximo) e orientação já conhecidos. Deste modo, a aplicação tem a navegação RA convencional e ao mesmo tempo tem a funcionalidade de reconhecimento de imagens executando e sendo mostradas ao usuário em paralelo, como mostrado na Figura 33 (d).

A Figura 37 mostra os passos geralmente presentes na abordagem baseada em visão (seta Imagem), na abordagem baseada em sensores (seta Sensores) e o controlador adaptativo que realiza a escolha entre cada abordagem.

Para cada *frame* obtido pela câmera, o processo é repetido para atualizar as informações virtuais apresentadas. A abordagem híbrida pode combinar o fluxo de imagem e sensores de diversas formas ou podem ser utilizados de forma intercalada.

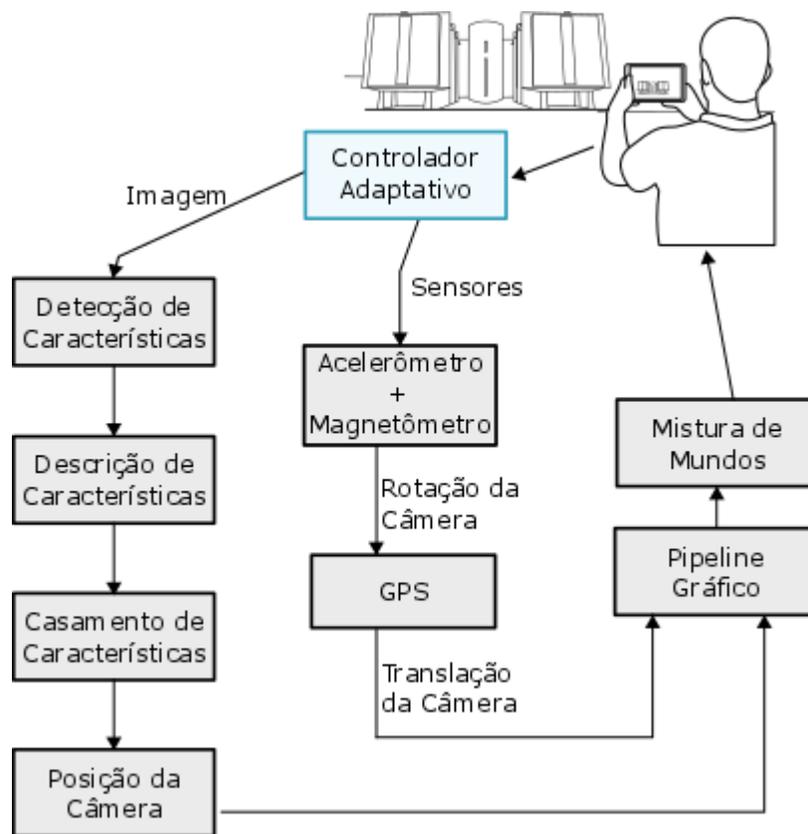


Figura 37. Pipeline do registro baseado em visão (seta Imagem) ao lado do registro baseado em sensores (seta Sensores).

Primeiramente foram testados os algoritmos de reconhecimento de imagens de forma isolada. Para essa avaliação de desempenho, foram testados os algoritmos ORB (Rublee et al., 2011),

BRISK (Leutenegger, Chli, Siegwart, 2011) e AKAZE (Alcantarilla, Nuevo, Bartoli, 2013) para detecção e descrição de pontos de interesse e os algoritmos RANSAC (Choi, 1981) e LMEDS (Rousseeuw, 1984) para o cálculo robusto da posição da câmera. Foram escolhidos esses algoritmos em razão deles possuírem descritores binários, que são mais rápidos para realizar o processamento e porque eles possuem diferentes formas de resolver o problema sendo ajustáveis a várias situações. Esses algoritmos estão descritos na Seção 4.6.

A Figura 38 mostra o gráfico comparativo entre o tempo e acurácia dos algoritmos utilizados no *pipeline* de reconhecimento de imagem. Estes foram testados de forma off-line (fora da aplicação) com imagens estáticas, com o intuito de obter um panorama que possibilite a escolha de quais algoritmos serão utilizados na implementação da aplicação protótipo.

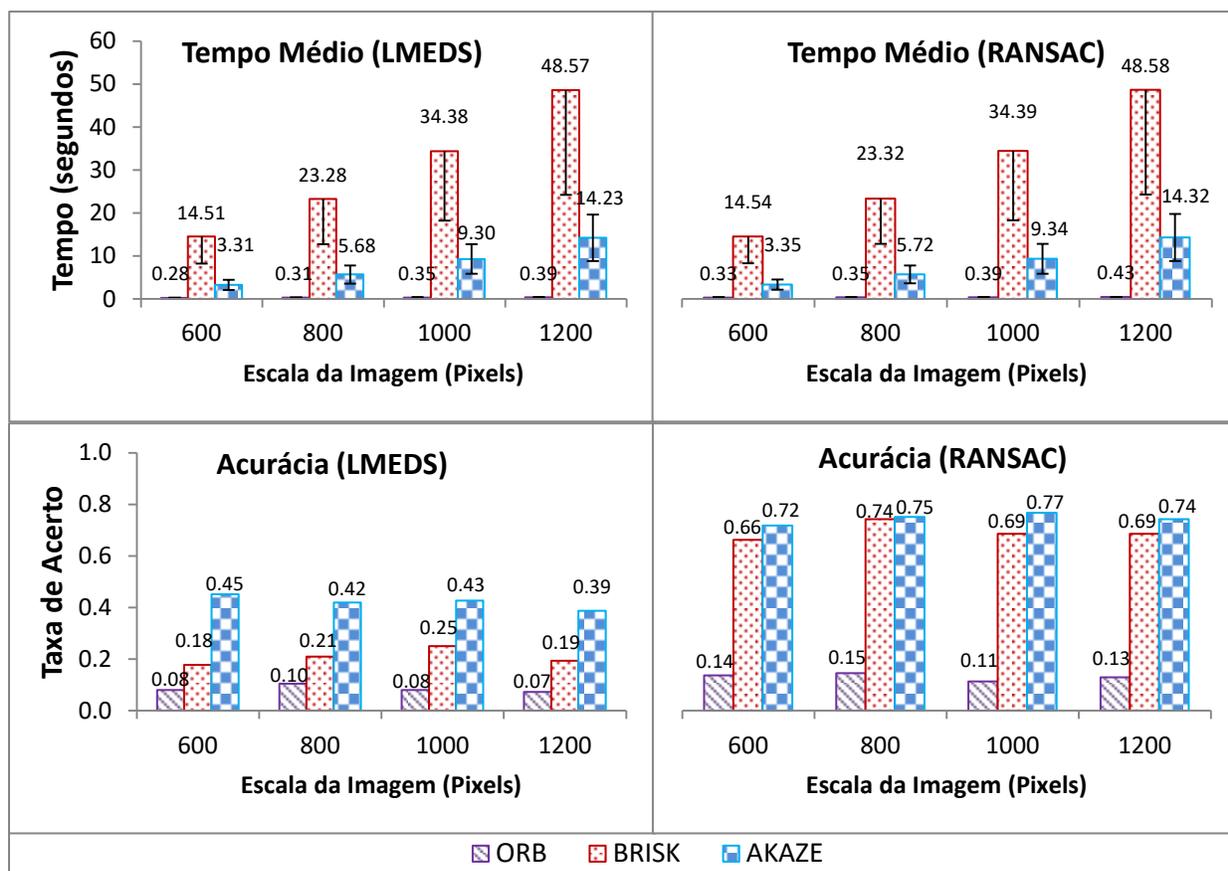


Figura 38. Gráficos dos testes de tempo e precisão mostrando os valores utilizados para escalar as imagens utilizadas nos teste e os algoritmos que foram comparados.

De acordo com os gráficos da Figura 38, é possível perceber que o algoritmo com melhor desempenho de tempo é o ORB com o RANSAC, apesar de sua acurácia ser muito baixa em relação aos outros algoritmos. Um algoritmo promissor é o AKAZE, pois sua acurácia é a

melhor, mas o tempo que este leva para ser executado é excessivo para aplicações de RAM em tempo real. Ainda de acordo com a Figura 38, o algoritmo RANSAC é melhor que o seu concorrente LMEDS em relação à acurácia e não tem tanta diferença comparando o tempo de execução. E o algoritmo BRISK tem um tempo e acurácia pior que o AKAZE.

É importante conhecer os algoritmos para combiná-los de acordo com suas características, com a finalidade de tornar o processamento mais rápido e acurado. Uma das contribuições do modelo apresentado nessa tese é a possibilidade da aplicação se adaptar aos diferentes contextos do usuário, dando a flexibilidade de escolha do algoritmo em tempo de execução. Essa flexibilidade na escolha do algoritmo ainda permite que as boas características de cada algoritmo sejam combinadas para gerar um reconhecedor melhor, assim como utilizar uma abordagem híbrida entre esses reconhecedores de imagem e sensores.

Por exemplo, ao utilizar o ORB de forma espaçada (utilizando os sensores para registrar o movimento entre as utilizações) temos um ganho de tempo considerável, como mostra a Figura 39, o qual foi gerado a partir da média de tempo medido em um dispositivo móvel. Para gerar os valores das médias no gráfico foram coletadas 100 medições de tempo para cada valor na abcissa, que representa o número de vezes que se deixa de executar o reconhecimento de imagem usando sensores como substituto.

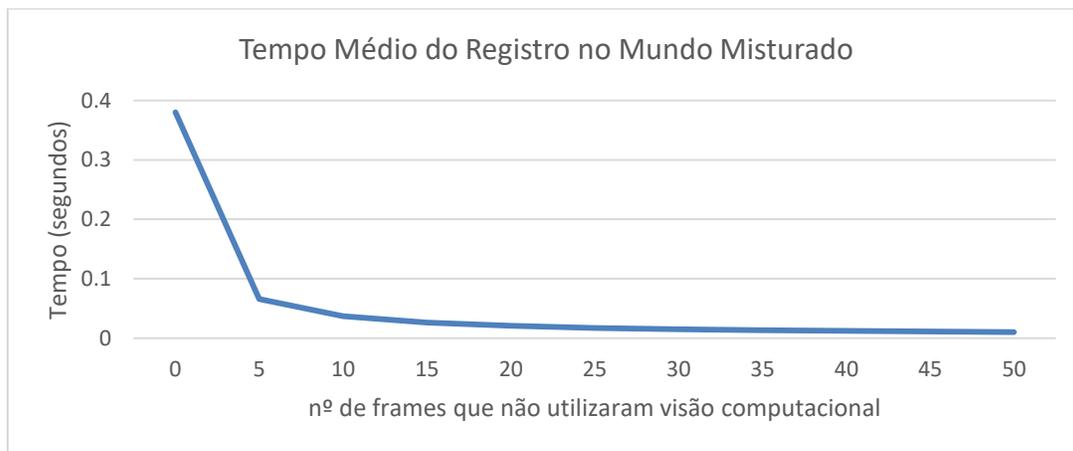


Figura 39. Gráfico do tempo médio para realizar o registro quando se utiliza de forma alternada reconhecimento de imagem e sensores. O valor das abcissas é número de nº de frames que utilizam sensores ao invés de reconhecimento de imagem.

Ainda que o tempo diminua consideravelmente à medida que se deixa de utilizar visão computacional, o registro sofre de problemas de alinhamento causados pela baixa acurácia dos

sensores (em relação aos objetos próximos). Assim deve-se considerar um equilíbrio na escolha do valor de *frames* a serem registrados por sensores ao invés de visão computacional. De acordo com o gráfico da Figura 39, um ponto de equilíbrio pode estar entre 5 e 15, uma vez que o tempo reduz muito pouco com mais de 15 *frames* e a precisão sofre por conta do uso contínuo de sensores. Para o desenvolvimento da aplicação de estudo de caso, foi utilizado o valor 10.

Apesar dos sensores terem uma acurácia ruim (se comparado a técnicas de visão computacional) em curtas distâncias, a utilização dos sensores nas aplicações de RAM é muito importante, porque dá uma noção precisa de localização e orientação em cenários que o usuário está longe dos POIs. Ainda assim, poucos estudos se dedicam a melhorar o desempenho dessa proposta.

4.8 Reconhecimento de Cena Interno-Externo

Quando o GPS não está com acurácia suficiente para determinar se o usuário está ou não dentro de um POI, a aplicação pode utilizar outros recursos para decidir se deve ou não mudar o modo de interação. Uma das possibilidades é o reconhecimento de imagens que pode ser utilizado para confirmar através das imagens da câmera se o usuário está em um ambiente externo ou interno.

No protótipo de validação foi implementado um classificador de ambientes interno/externo que utiliza o histograma individual de cada camada de cor do modelo YCrCb contendo 8 intervalos igualmente distribuídos para cada camada. Uma vez que trabalhos mostram que dividir a imagem em vários pedaços e classificar esses pedaços individualmente gera melhores resultados (Szummer e Picard, 1998), (Serrano et al., 2002), (Kim et al., 2010), (Payne e Singh, 2005), a classificação implementada divide a imagem em 16 pedaços igualmente espaçados.

O treinamento foi realizado com bases de dados públicas (1800 imagens de cada classe: interno e externo) e foi validado com imagens retiradas do ambiente de teste (25 imagens de cada classe). A extração das características para o treinamento foi realizado por um código feito na linguagem *python* com a utilização da biblioteca OpenCV, que extraiu e salvou as características de cada imagem juntamente com a classe.

Para gerar o modelo de predição foi utilizado o software Weka. Após alguns testes, foi verificado que a utilização da técnica *Random Forest* obteve bons resultados (82% de acertos)

com validação cruzada de 10 *folds*. Depois de treinado esse modelo foi exportado para ser lido pela aplicação que funciona no sistema Android. A validação com o ambiente de teste revelou uma taxa de acerto de 77%.

O mesmo algoritmo de extração de características foi implementado na aplicação RAM, mas com a imagem capturada pela câmera do dispositivo, e o modelo treinado foi importado para realizar a predição com base nas características extraídas da imagem da câmera.

5. AVALIAÇÃO DA APLICAÇÃO

Este capítulo apresenta a avaliação da aplicação protótipo que valida o modelo adaptativo proposto nesta tese.

5.1 Organização do Teste

O teste de avaliação foi realizado através de entrevistas com 11 participantes que utilizaram a aplicação protótipo em um ambiente real. Antes da realização da entrevista, o participante foi convidado a seguir um percurso predefinido para uniformizar o teste. O percurso teve início no setor profissional da UFPA, próximo ao Atelier de Artes (está a aproximadamente 700 metros do POI alvo) com término na sala do LABVIS que fica localizado no segundo andar do prédio do ICEN (a posição hierárquica de cada local foi explicada na seção 4.2).

Como critério para a escolha dos participantes do teste, foi definido que eles deveriam ter formação em ciência da computação ou engenharia da computação, tendo em vista que eles podem opinar com mais propriedade sobre o comportamento do sistema, além de dar sugestões mais técnicas de melhorias. Durante a realização do percurso e da entrevista foi solicitado que o participante tentasse se colocar no lugar de quem iria utilizar a aplicação, ou seja, que opinassem pensando em usuários de público mais geral.

No início do percurso foi solicitado que o participante traçasse uma rota até o LABVIS. Em cada momento de transição também foi solicitado que o participante parasse em um ponto e descobrisse os lugares ao seu redor.

Dentro do ICEN foi solicitado aos participantes que realizassem o reconhecimento de placas de salas de aula para descobrir informações sobre os lugares, como por exemplo, o cronograma de aulas, lista de e-mails e a agenda de atividades extras.

Logo após a realização do percurso foi realizada a entrevista, que foi gravada em áudio para análise posterior. Para realizar a entrevista foram definidas perguntas como forma de roteiro, mas que foram feitas ao longo da entrevista de maneira a seguir o fluxo de uma conversa, sempre orientando o usuário quando o mesmo não compreendesse o que era proposto na pergunta. Ou seja, as perguntas serviram como um roteiro para o entrevistador seguir um fluxo e não esquecer nenhuma pergunta. A Tabela 3 mostra as 14 perguntas presentes no roteiro.

Tabela 3. Roteiro de perguntas feitas aos participantes do teste.

1	A aplicação te ajudou a saber onde você estava?
2	Você se perdeu em algum momento? Em que situação isso ocorreu?
3	Você considera que os marcadores virtuais estavam posicionados no lugar correto? Houve diferença no posicionamento quando em ambientes interno e externo?
4	Você percebeu se houve mudanças na aplicação? Quando? Quais?
5	Visualmente falando, os marcadores dão a entender que estão atrelados aos locais ou que estão flutuando na tela sem relação com o ambiente?
6	Quais os pontos positivos e negativos de explorar conteúdos dessa forma?
7	A aplicação auxiliou na descoberta de itens que podem ser explorados? Melhorou a sua percepção sobre esses itens? Existe alguma particularidade dessa melhora (caso exista) para ambientes externos ou internos?
8	Você pode dar alguns exemplos de onde mais essa aplicação pode ser usada?
9	Você considera que a aplicação se adaptou (se ajustou) ao ambiente que você está?
10	O reconhecimento de imagem foi útil para explorar conteúdo?
11	Quais os pontos positivos e negativos da aplicação apresentar mais informações específicas à medida você se aproxima ou entra nos lugares?
12	Você prefere ter todas as informações disponíveis ao mesmo tempo (utilizando um filtro por distância, por exemplo) ou receber informações associadas aos lugares, de forma gradual, enquanto vai se aproximando ou chega a um lugar?
13	O que você achou do mecanismo de busca? Você tem críticas ou sugestões?
14	De maneira geral, em que aspectos a aplicação pode melhorar?

Além da entrevista realizada com participante, o trajeto realizado durante a fase de experimentação foi registrado através de *logs* da aplicação. Uma vez que a aplicação registra a posição que o sistema calculou, que pode não ser a posição correta, outro avaliador registrou o trajeto mais próximo do verdadeiro utilizando outra aplicação (específica para esse propósito), possibilitando uma forma de comparar e avaliar a posição que o sistema calcula para o usuário.

Inicialmente foi realizado um teste piloto com um participante para treinamento dos realizadores do teste e para correção de eventuais *bugs* da aplicação, sendo que este teste piloto não foi considerado para a análise dos resultados. O teste foi realizado com onze participantes todos formados em curso superior na área de tecnologia, sendo que todos possuem experiência na utilização de dispositivos móveis.

O protocolo *Think-Aloud* (Fonteyn, 1993) foi iniciado no teste piloto, mas foi percebido que a bateria utilizada no gravador, que geralmente ficava próximo do *smartphone* utilizado no teste, gerava uma interferência nos sensores do dispositivo móvel, prejudicando o posicionamento dos marcadores virtuais na tela. Sendo assim, o protocolo *Think-Aloud* foi descartado para condução do teste.

5.2 Registro do Percurso do Teste

A Figura 40 mostra uma visão geral de todos os percursos realizados pelos participantes. As linhas azuis mostram o percurso registrado por um condutor do teste e representa de forma mais precisa qual foi o caminho realizado pelo participante. As linhas vermelhas representam a posição registrada pela aplicação, posição essa utilizada para calcular o posicionamento dos marcadores virtuais apresentados para o usuário.

Ainda segundo a Figura 40 é possível perceber que ocorreram dois momentos de falha crítica no posicionamento (B), que são os dois pontos vermelhos que se distanciam bastante do trajeto azul. No geral, as linhas vermelhas mostram que apesar do posicionamento via GPS ter uma boa acurácia, este apresenta uma precisão ruim. Essa precisão tem impactos maiores no posicionamento dos marcadores virtuais dos POIs que estão próximos ao usuário, pois essa variabilidade se torna mais notável em curtas distâncias. Um momento em que a acurácia não está boa é no início do trajeto (do lado direito da ponte), possivelmente porque o sistema de GPS demanda um tempo para realizar a triangulação com os satélites que fornecem esse serviço.

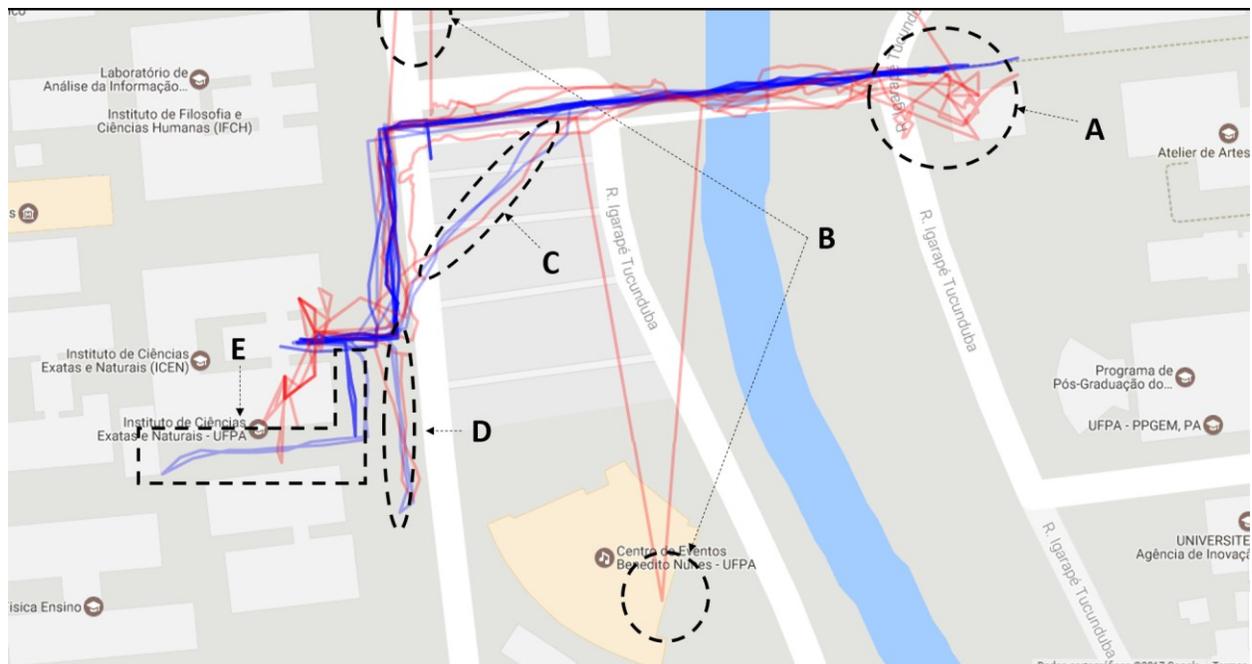


Figura 40. Visão geral dos caminhos registrados pela aplicação medida por GPS, durante o trajeto realizado pelos participantes (linhas vermelhas) em comparação com os caminhos registrados manualmente por um condutor do teste (linha azul). O caminho registrado pelo condutor do teste serve de referência.

Dois participantes decidiram cortar caminho pelo estacionamento EM, mas apenas um foi devidamente registrado por falhas no registro. Adicionalmente, outros dois participantes perceberam que o marcador não estava apontando na direção correta do prédio do ICEN, quando chegaram próximo do ICEN, e resolveram seguir o caminho erroneamente sugerido para verificar se a aplicação corrigia o problema (D) (E). Um dos casos a correção foi sanada (D), e constata-se pelo fato da linha vermelha acompanhar a linha azul, mas no outro caso a correção não foi feita pela aplicação (E).

Para melhor visualização, a Figura 41 (a) destaca o trajeto de um dos participantes. Percebe-se que neste caso os maiores problemas de posicionamento ocorreram nas extremidades do percurso, ou seja, onde o trajeto começa, e próximo ao prédio do ICEN. A Figura 41 (b) mostra as falhas que ocorreram com outro participante em dois momentos: ao cruzar a ponte, onde a localização foi reajustada para um ponto distante do percurso; e ao chegar ao prédio do ICEN, onde a aplicação sugeria que o prédio estivesse mais à frente.

Quando o participante entra no prédio do ICEN é iniciada a localização interna e da mesma forma que no ambiente externo, marcadores virtuais são apresentados para ele, com a diferença de que agora estes marcadores indicam IOIs dentro do prédio. Para avaliar a localização interna

também foram registradas as localizações dadas pelo dispositivo, calculadas através da potência dos sinais Wi-Fi, além disso, o condutor do teste continuou registrando o caminho dos participantes em uma aplicação separada desenvolvida especialmente para este propósito.

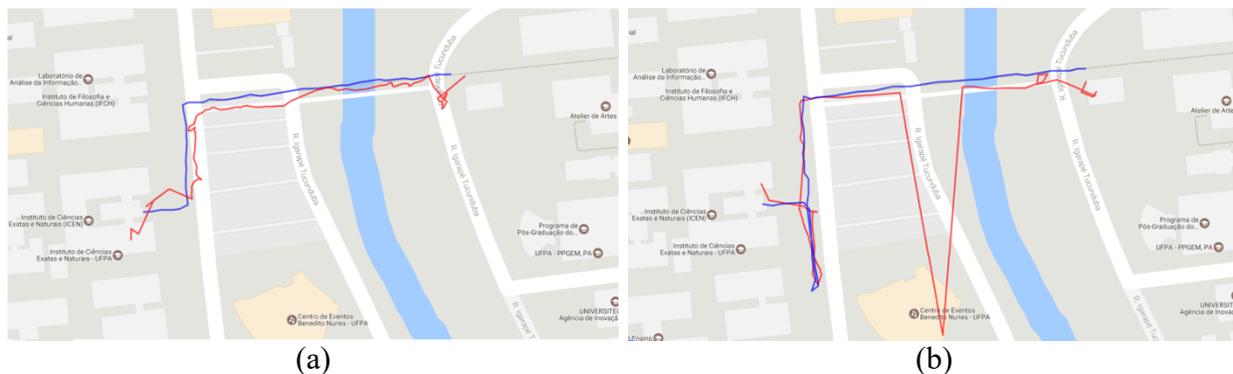


Figura 41. Destaque do trajeto realizado por dois participantes do teste. Nestes é possível observar uma maior variação nas extremidades da localização e uma falha no posicionamento do GPS (b).

A Figura 42 mostra a visão geral do caminho feito pelos participantes dentro do prédio do ICEN. No primeiro andar (esquerda da Figura 42) foi solicitado aos participantes que não conheciam o laboratório da pós-graduação em ciência da computação que tentasse encontrar o mesmo dentro do ICEN. Caso o participante já conhecesse o laboratório, nada era solicitado e o mesmo dirigia-se naturalmente até a sala do LABVIS no segundo andar.

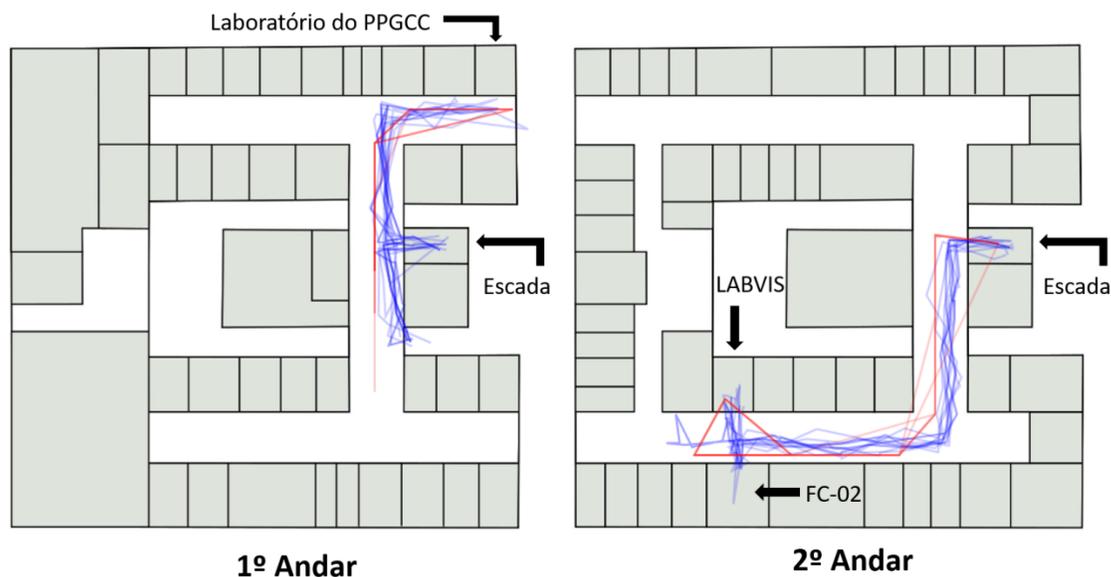


Figura 42. Visão geral do caminho feito pelos participantes dentro do prédio do ICEN. As linhas vermelhas representam a posição dos participantes obtidas pela localização com sinais de Wi-Fi e as linhas azul foram registradas por um condutor do teste.

Percebe-se que a localização interna tem uma boa acurácia e tem alta precisão, uma vez que os pontos treinados pelo algoritmo eram discretos. A Figura 43 destaca o trajeto de um participante e mostra que apesar da boa acurácia e alta precisão, a localização interna sofre de atrasos para atualizar a posição e isso causa uma dessincronização de tempo entre a posição atual do usuário e a medida pelo sistema. Esse caso é destacado no ponto (A) da Figura 43 (segundo andar), onde ocorre um salto da posição (B) que era no primeiro andar próximo a escada para esse ponto (A) que fica no cruzamento de dois corredores.

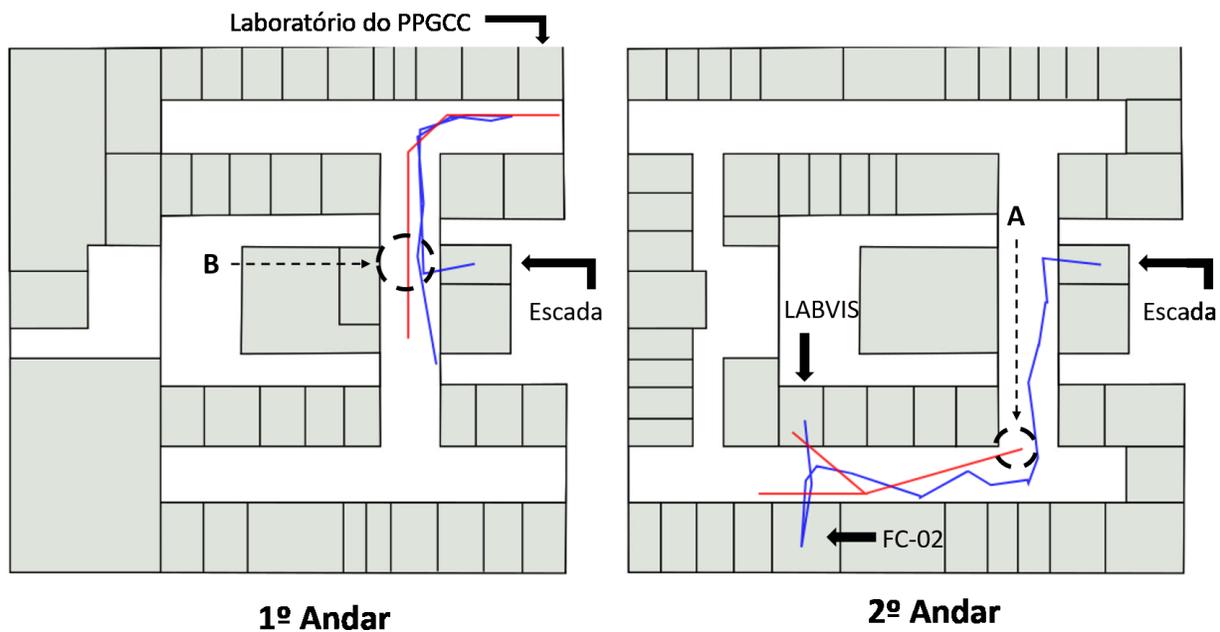


Figura 43. Destaque do caminho de um participante dentro do prédio do ICEN, mostrando o momento em que o participante sobe a escada (B) e ocorre uma demora em atualizar sua localização, fazendo com que a aplicação dê um salto com a localização do usuário.

Adicionalmente foram registrados os dados de posicionamento do GPS mesmo dentro do prédio (esses dados não eram usados no cálculo de localização interna) para efeitos de comparação com a localização interna. A Figura 44 destaca os dados e revela que seria inviável utilizar esses dados para realizar a mesma tarefa que foi realizada com a localização interna.

A visualização revela que alguns dos assuntos pertinentes são: a aplicação, a localização, o marcador, o ambiente, a percepção, a mudança e assim por diante. Dessa forma, foram destacadas algumas frases (escolhidas pela distinção das ideias) que continham essas palavras.

A Figura 46 apresenta as frases relacionadas com a palavra “marcador”. As frases revelam que existiu uma latência para atualização da posição correta dos marcadores virtuais, que o erro no posicionamento dos marcadores pode ter sido motivado por falhas no GPS, que a grande quantidade de marcadores gera um problema de oclusão deixando a visualização confusa, que em ambientes externos o posicionamento dos marcadores estava pior que internamente, que os marcadores da forma como foram desenhados não dão a impressão de realismo e parecem estar flutuando na tela, e que o reconhecimento de imagem torna a associação entre o marcador e o POI mais fácil.



Figura 46. Destaque de frases relacionadas com a palavra “marcador”.

A Figura 47 apresenta as frases relacionadas com a palavra “localização”. As frases revelam que existe uma latência para atualizar a localização, que essa latência pode atrapalhar o usuário, que é possível se localizar apesar dos erros do GPS, que a localização hierárquica ajuda na própria localização, que houve erros no posicionamento do andar em ambientes internos e que o radar muda conforme a localização do usuário.



Figura 47. Destaque de frases relacionadas com a palavra “localização”.

A Figura 48 apresenta as frases relacionadas com a palavra “perceber”. As frases apontam que a direção e a adição de novos pontos são percebidas durante a caminhada, que há uma contradição quanto aos mecanismos atuais para percepção das mudanças de ambiente, que é difícil perceber pontos quando há grande quantidade e que mudança de andar e de externo para interno são mais perceptíveis.

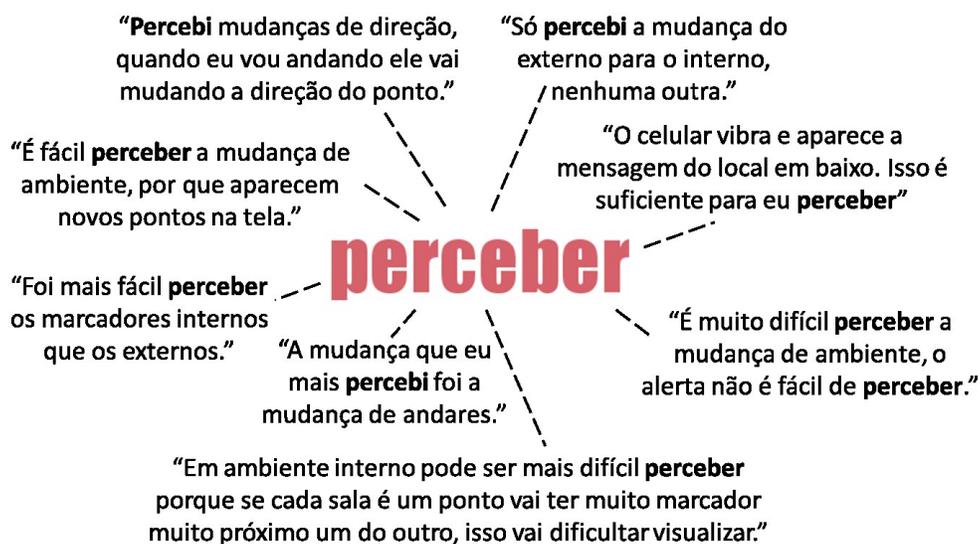


Figura 48. Destaque de frases relacionadas com a palavra “perceber”.

A Figura 49 apresenta as frases relacionadas com a palavra “mudança”. As frases indicam que as mudanças foram percebidas no decorrer do caminho, que a atualização no radar é uma boa mudança, que a mudança de ambiente não foi percebida, que a mudança dos marcadores na

tela é percebida pelos participantes e que poderia ter outras formas de indicar mudanças, como avisos sonoros.



Figura 49. Destaque de frases relacionadas com a palavra “mudança”.

Todas as frases extraídas dos áudios das entrevistas foram codificadas na qualificação de comentários (positivo, negativo ou neutro), qual a atividade que a frase se refere, se algum problema foi identificado e qual. A escala Likert foi utilizada para qualificação dos comentários e foi definida com base no conteúdo da frase, sendo que quando há somente pontos positivos na frase é dado o valor 5, quando há majoritariamente pontos positivos com uma ou mais ressalvas negativas é dado o valor 4, quando a frase dá sugestões e pontos de vista sem apontar pontos positivos e negativos do mecanismo atual é dado o valor 3, quando há majoritariamente pontos negativos com uma ou mais ressalvas positivas é dado o valor 2 e quando há somente pontos negativos é dado o valor 1.

A Figura 50 mostra um gráfico que relaciona a qualificação da frase com a atividade a que ela se refere. A cor se indica a qualificação que pode ser positivo, negativo ou neutro e o comprimento das seções verticais indica a proporção da quantidade de frases classificadas em cada grupo. A partir desse gráfico pode-se perceber que a maioria das qualificações expostas na entrevista foram positivas (~50%), seguida pelas frases com qualificações negativas (~32%) e algumas frases com qualificação neutra (~18%).

A descoberta de POI foi a atividade mais comentada (~22%) e obteve mais comentários positivos (~49%) e neutros (~33%) que negativos (~18%), o que mostra que a ferramenta tem

papel fundamental na descoberta de novos POIs e consequentemente na percepção do usuário sobre os seus arredores.

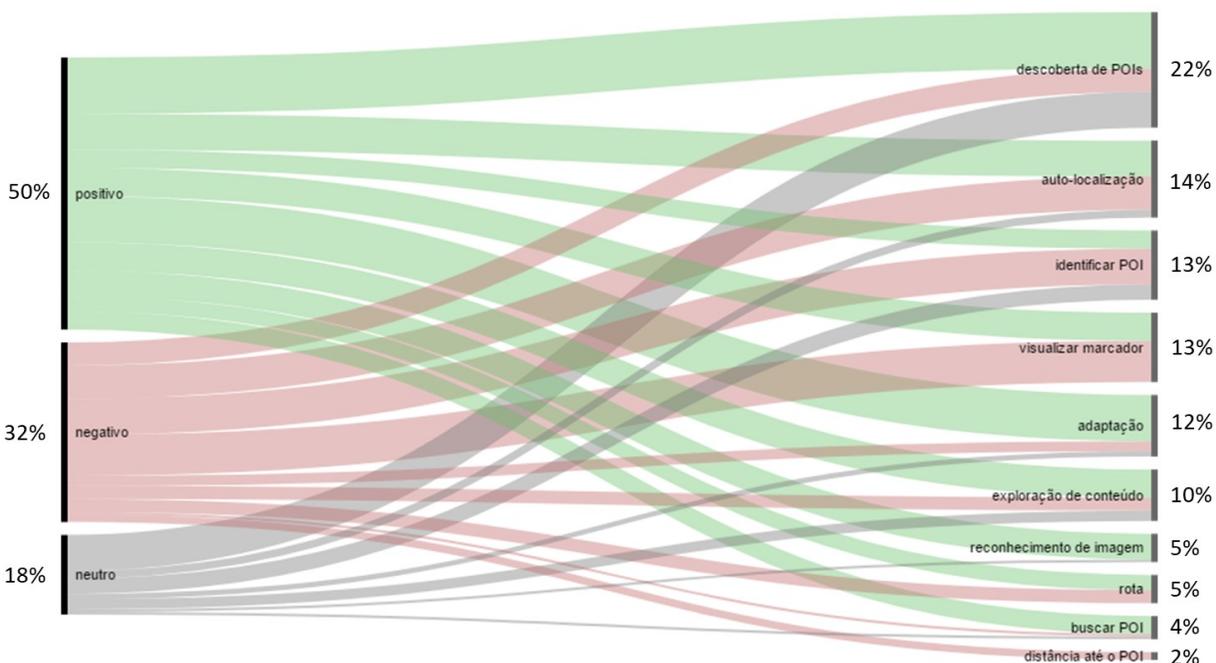


Figura 50. Relacionamento da qualificação das frases (eixo esquerdo) em relação às atividades que as frases se referiram (eixo direito).

A auto-localização foi a segunda atividade mais comentada (~14%) e os comentários sobre a mesma foram divididos, obtendo ~47% de comentário positivos, 43% de comentários negativos e ~10% de comentários neutros, o que mostra que apesar da localização do usuário ter uma boa resposta existe alguns problemas a serem resolvidos.

As tarefas de identificar POI e visualizar marcador (ou visual do marcador) foram as únicas que tiveram maior proporção de comentários negativos, totalizando ~52% e ~55% respectivamente. Sendo que identificar POI foi a que teve menor proporção de comentários positivos (~26%) e obteve ~22% de comentários neutros, sendo que o principal impacto na identificação do POI é o posicionamento inadequado dos marcadores, que por sua vez é causado pela baixa acurácia do sistema de posicionamento.

Já a tarefa de visualizar marcador teve opiniões mais divididas com ~41% de comentários positivos e ~4% de comentários neutros, uma vez que alguns participantes gostaram do visual do sistema e outros acharam que não, alguns tiveram a impressão que o marcador estava

visualmente atrelado ao POI físico e outros tiveram a sensação de que o marcador estava deslizando na tela.

As funcionalidades de adaptação e reconhecimento de imagens obtiveram maior proporção de comentários positivos (~67% e ~91% respectivamente). A adaptação aos ambientes é o principal foco desta tese, e embora isso não tenha sido comentado com os participantes do teste, esta foi percebida majoritariamente como algo positivo, obtendo ~17% de comentários neutros e ~16% de comentários negativos. Já o reconhecimento de imagens não obteve comentários neutros e teve apenas 9% de comentários negativos.

Outras duas atividades que merecem destaque por comentários positivos é a busca de POIs e a exploração de conteúdos através da RA. A rota teve comentários divididos entre positivo e negativo. Enquanto que a distância até o POI obteve dois comentários negativos porque a mesma não havia sido implementada no primeiro e segundo teste, sendo que a partir do terceiro teste a distância era mostrada juntamente com a legenda do marcador, o que foi suficiente para não gerar mais comentários sobre essa questão.

Com o objetivo de entender melhor o problema que gerou os comentários negativos foram codificados os principais problemas presentes nas falas dos participantes da entrevista. A Figura 51 mostra o gráfico que relaciona as atividades com os problemas presentes nos comentários. Analisando o eixo respectivo aos problemas (eixo direito), observa-se que os principais problemas (em ordem decrescente) que podem influenciar a experiência dos participantes durante o teste foi a demora na atualização da aplicação, as falhas do GPS, a falta de informações sobre os POIs e problemas de design (estrutural e visual) de interface gráfica.

A identificação de POI foi prejudicada principalmente pela demora na atualização, falhas no GPS, precisão do sistema (posicionamento correto dos marcadores) e pela oclusão de POIs sobrepostos. A visualização dos marcadores foi atribuída às falhas do GPS majoritariamente e tiveram vários outros problemas envolvidos com menor proporção.

A auto-localização teve problemas principalmente com a demora na atualização da aplicação e com falhas do GPS, em menor proporção revelam problemas de falta de informação e falhas na localização interna. Os problemas relacionados com a adaptação foram principalmente pela demora na atualização e por problemas de interface gráfica.

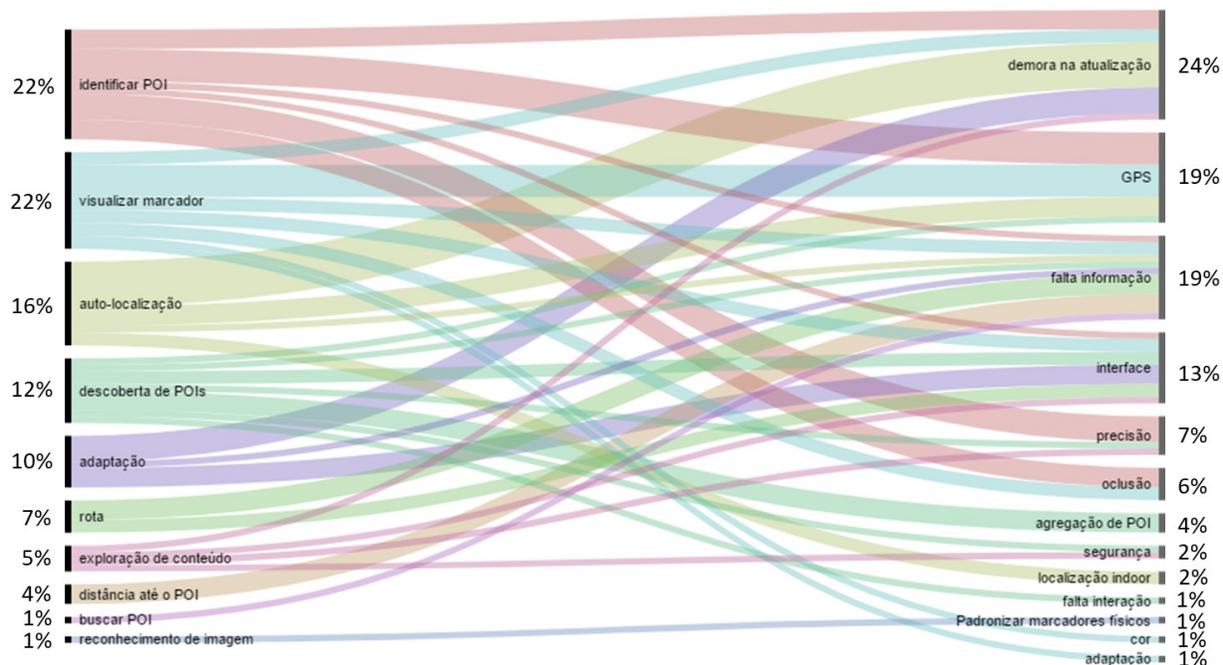


Figura 51. Relacionamento das atividades que as frases dos participantes se referem (eixo esquerdo) em relação aos problemas observados (eixo direito). Essa visualização mostra apenas as frases que comentavam sobre um ou mais problemas.

Como explicado anteriormente, para cada frase obtida das entrevistas foi dado uma nota dentro da escala Likert para dar um maior detalhamento da qualificação dos comentários. A Figura 52 apresenta o gráfico de barras da quantidade de comentários dentro da escala Likert que estão relacionados com a adaptação da aplicação. O gráfico confirma que a adaptação foi na maioria dos casos vista como algo positivo e mostra com mais detalhes essa informação, com comentários de exemplo do lado direito da barra.

Os comentários positivos sobre a adaptação da aplicação ressaltaram que é fácil perceber as alterações, mas que houve um atraso para essa adaptação ocorrer, isso devido às limitações de hardware do dispositivo móvel utilizados no teste e porque o posicionamento interno está sendo calculado dentro da aplicação com o uso de aprendizado de máquina, o que demanda tempo de execução. Observa-se que há uma percepção para a adaptação da aplicação entre interno e externo uma vez que essa transição gera adaptações no método de posicionamento do usuário e na interface gráfica, como é o caso do radar.

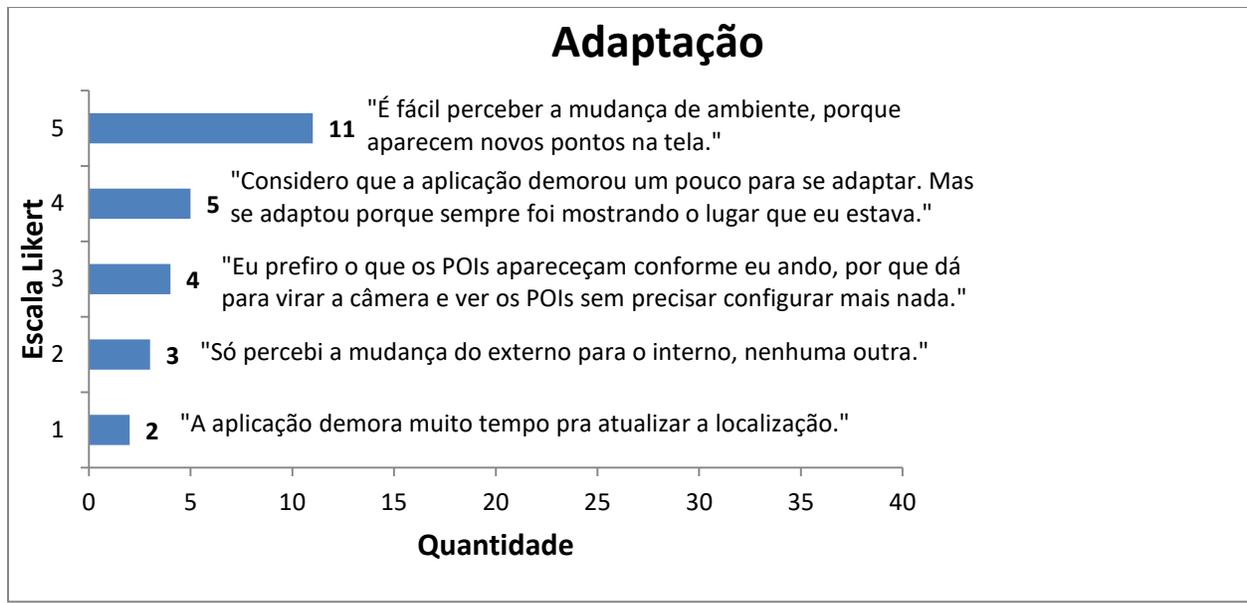


Figura 52. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

A Figura 53 mostra o gráfico da escala Likert relacionada com auto-localização que a ferramenta provê. Observa-se que os comentários estão bem divididos e que os problemas que originaram os comentários negativos foram relacionados com a demora na obtenção da localização e as falhas do GPS. Esse gráfico mostra que apesar de existirem problemas com auto-localização decorrentes da baixa acurácia do GPS, os participantes conseguiram se localizar com o destaque para o posicionamento hierárquico.

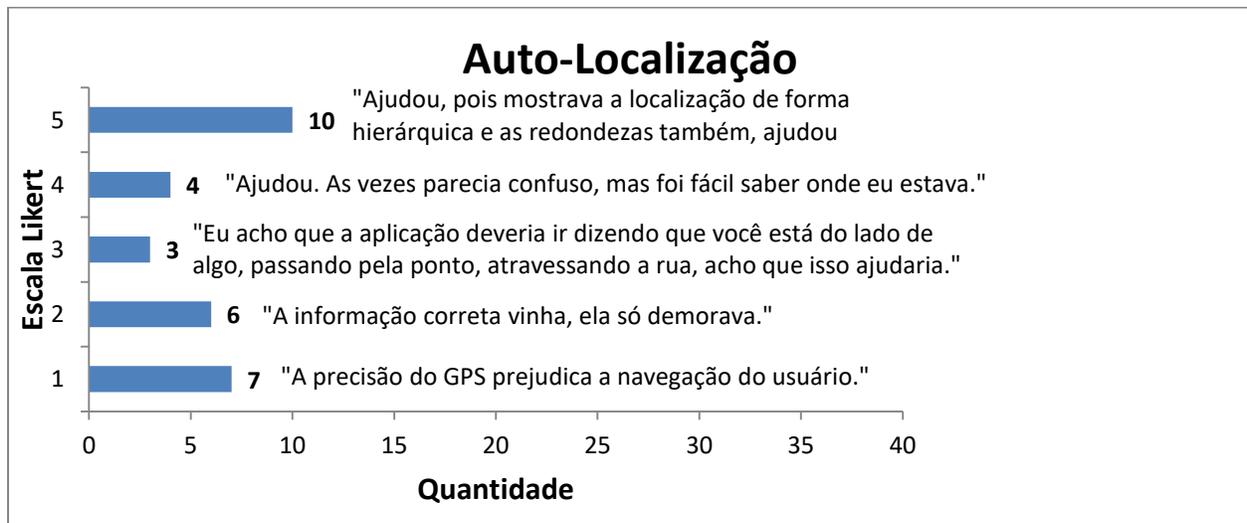


Figura 53. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

De acordo com a Figura 54 a descoberta de POIs foi vista positivamente pelos participantes do teste, mostrando que a ferramenta tornou mais fácil e intuitivo o processo de descoberta de POIs ao redor do usuário. Os comentários neutros concentraram principalmente os exemplos de onde a aplicação pode ser usada (sendo turismo o principal exemplo de uso), e como pontos negativos foram levantadas questões de segurança (tanto da informação quando física) e melhorias na precisão e desempenho da aplicação.

Ressalta-se a importância da adaptação para descoberta de POIs quando deram o exemplo do turismo, uma vez que o turista deseja saber mais sobre o local que está visitando, desde o momento que se escolhe para onde deseja ir até quando chega ao local e deseja saber o que esse local oferece.

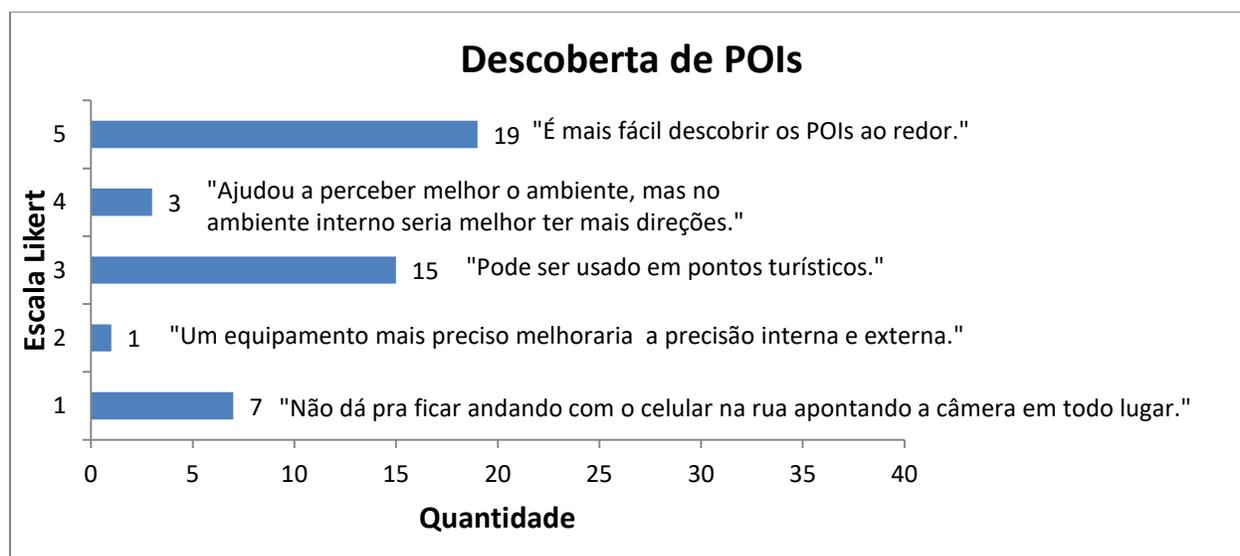


Figura 54. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

Analisando o gráfico da Figura 55 percebe-se que os participantes tiveram problemas com a identificação de POIs e dois dos motivos citados foram a grande quantidade de POIs, o que gera sobreposição de marcadores e dificulta a legibilidade, e a confusão do posicionamento dos marcadores, que quando próximo ao prédio do ICEN apontavam pra outro lugar que não o prédio correto. Como visto anteriormente nos resultados do registro da localização, quando se chega próximo ao prédio o problema da baixa precisão do GPS se torna mais evidente.

Posto que a precisão do GPS tem impacto direto na identificação dos POIs, a adaptação do método de localização é relevante para tornar viável a identificação de POIs em lugares que o GPS geralmente falha, como é o caso de ambientes internos.

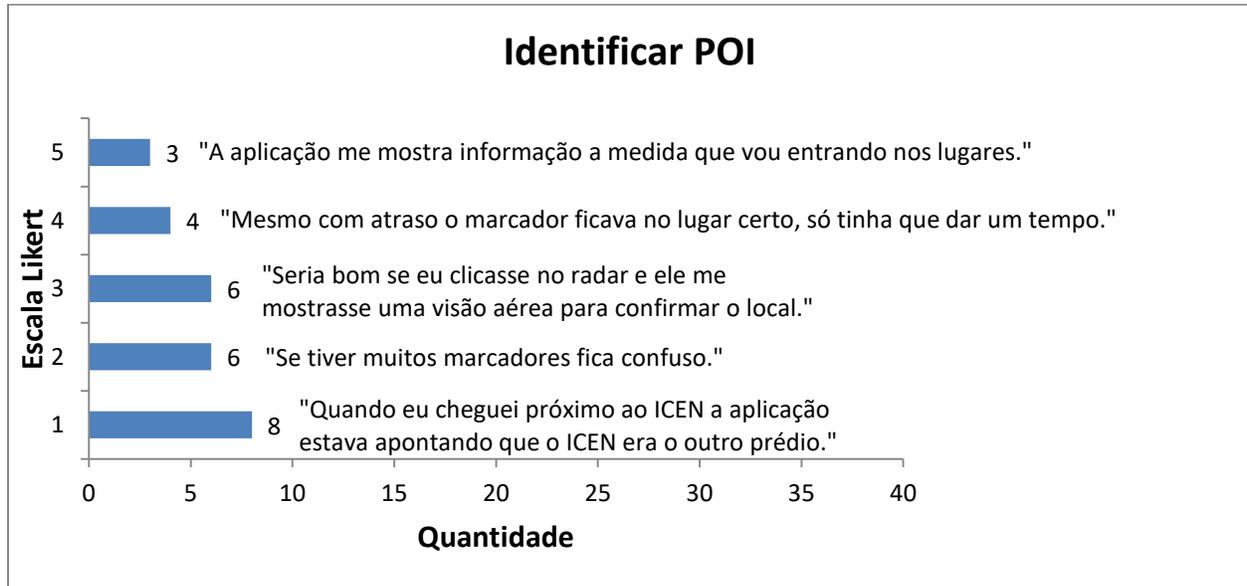


Figura 55. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

De acordo com a Figura 56 os participantes opinaram de forma dividida em relação ao aspecto visual do marcador, sendo que alguns participantes tiveram a sensação de presença do marcador no ambiente e outros tiveram a impressão que ele simplesmente estava flutuando na tela.

Apesar do visual estético do marcador ainda precisar de melhorias, a adaptação auxilia no desenho mais adequado para o marcador em cada situação, por exemplo foi comentada a seguinte frase durante a entrevista "Quando reconhece uma imagem fica muito mais fácil de associar o marcador a sala.", sendo assim a adaptação do registro ajuda o usuário na identificação do POI ou IOI alvo e melhora o aspecto visual do marcador pois com o reconhecimento de imagens o marcador é renderizado de forma mais alinhada ao POI ou IOI.

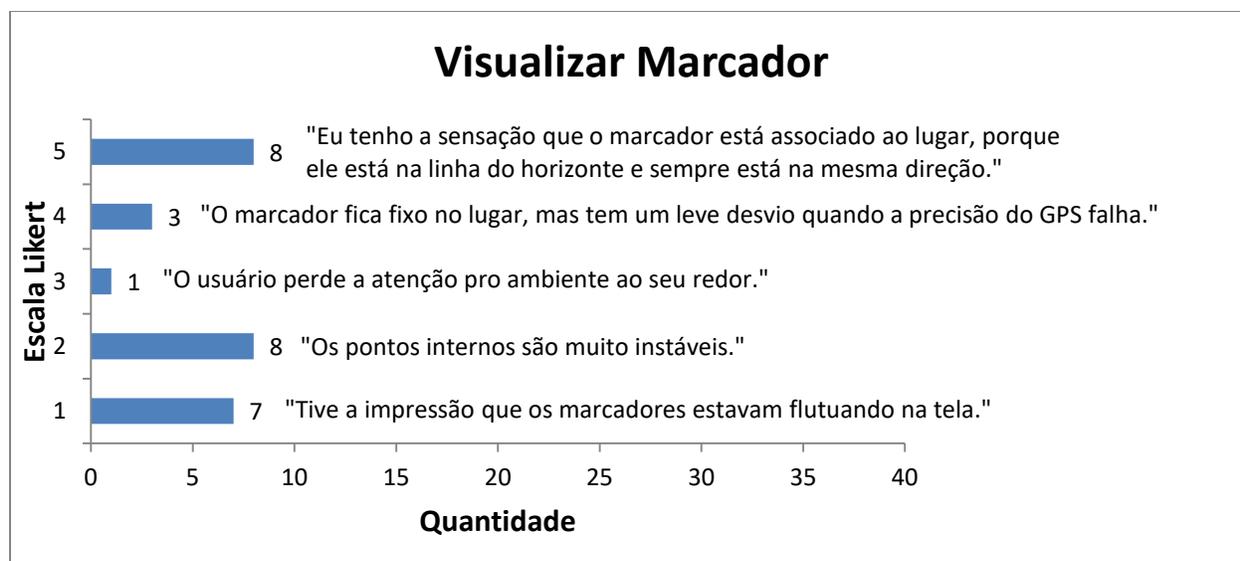


Figura 56. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

A Figura 57 mostra o gráfico da escala Likert relacionado com a busca de POIs que foi feita pela inserção textual de termos de busca. A busca de POI foi relatada como essencial nessa aplicação, principalmente pelo fato de permitir que sejam encontrados lugares que não aparecem na tela do usuário devido a agregação de POIs. Uma sugestão foi para que seja inserido a busca por comandos de voz e outro comentário sugere que se tenha a lista de todos os POIs de antemão à busca textual.

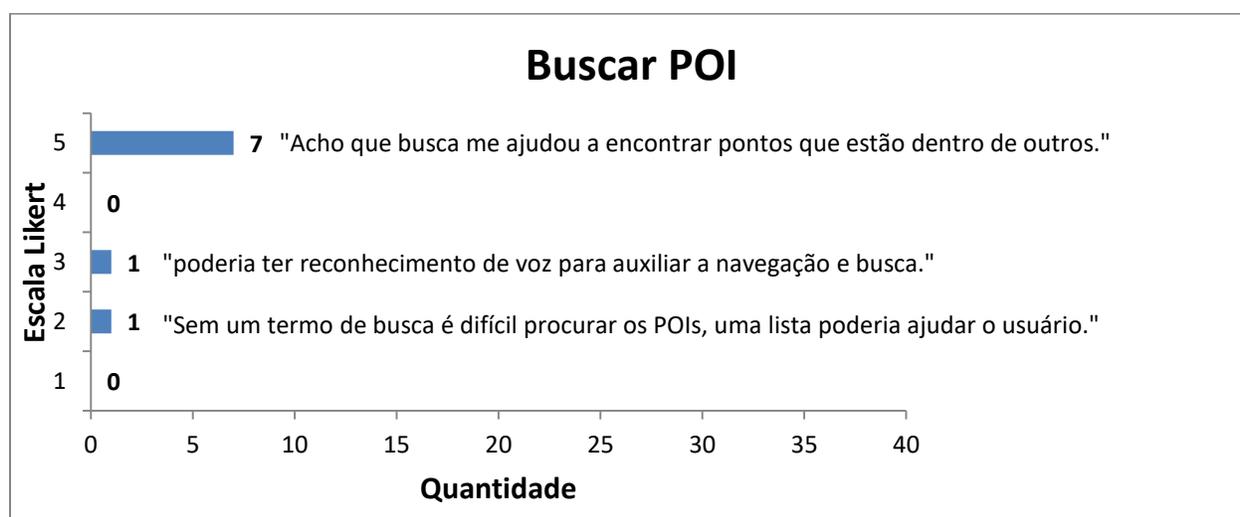


Figura 57. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

Quanto à rota, a Figura 58 mostra que quatro comentários ressaltaram os aspectos positivos da mesma, que conduz o usuário passo a passo até o seu destino, mostrando como se entra e sai dos lugares organizados de forma hierárquica. Alguns problemas de interface gráfica foram relatados, como falta de mensagens fixas e a falta de indicação no radar.

A adaptação da aplicação ajuda o usuário a seguir uma rota desde o ambiente externo até o local desejado, mesmo que esse local esteja dentro de prédio por exemplo. Alguns participantes sugeriram que a aplicação poderia ir informando para o usuário a cada momento em que local o mesmo se encontra e poderia ir informando os passos que o mesmo deve realizar para chegar até o destino, como o seguinte comentário “Eu acho que a aplicação deveria ir dizendo que você está do lado de algo, passando pelo ponto, atravessando a rua, acho que isso ajudaria.”. Da mesma forma, quando o usuário entra em um prédio e deseja ir à uma sala a aplicação pode dar a rota em forma de comandos a serem seguidos, como dobre a direita, suba a escada e entre na terceira porta à esquerda.

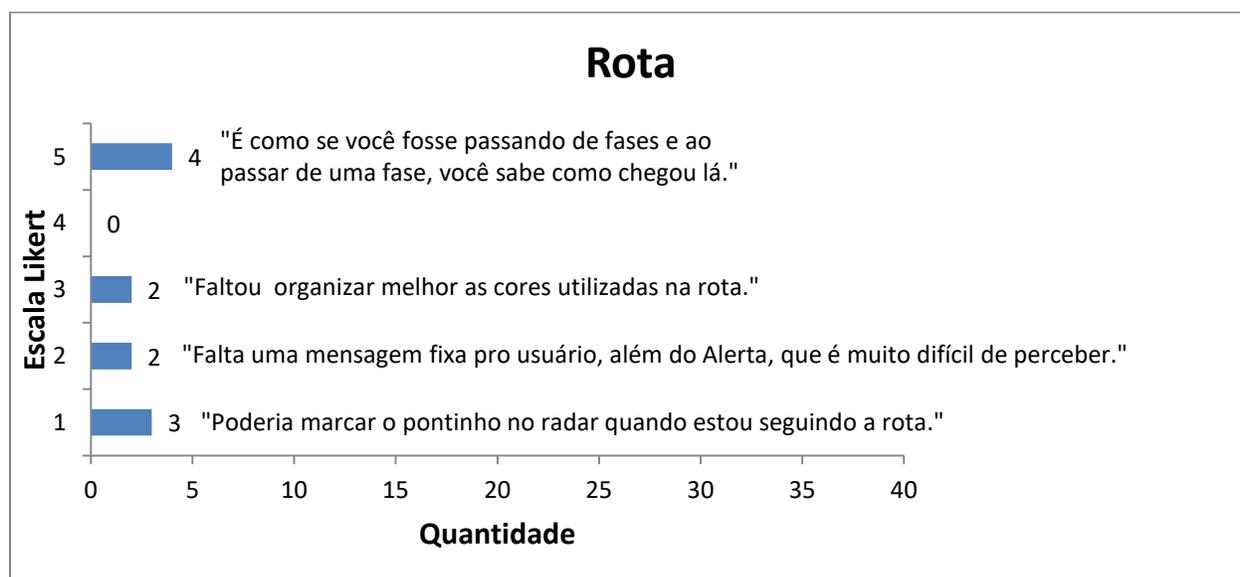


Figura 58. Escala Likert dos comentários sobre a funcionalidade de adaptação com exemplos de frases em cada nível da escala.

CONCLUSÃO

A presente tese de doutoramento versa sobre um modelo adaptativo às transições do usuário em ambientes internos e externos para aplicações de RAM, com objetivo de permitir ao usuário experiências semelhantes tanto em ambiente interno quanto externo usando a mesma aplicação. O processo de adaptação está relacionado com a mudança da aplicação em:

- Nível de dados, selecionado e exibindo somente os dados que estão próximos ao usuário;
- Forma como os marcadores virtuais são apresentados para o usuário;
- Adaptação da interface, principalmente no aspecto do radar para descoberta de POIs ou IOIs;
- Utilização de técnicas de reconhecimento de imagens de acordo com a posição e orientação do usuário;
- Troca automática do método de rastreamento de dispositivo móvel;
- Rastreamento híbrido com escolha dos algoritmos de reconhecimento de imagem e a utilização do mesmo de forma alternada com sensores.

O modelo arquitetural adaptativo às transições entre ambientes internos e externos está situado como mais um passo dentro do *pipeline* de aplicações de RAM e apresenta o uso combinado de técnicas de localização do usuário através sensores de curto e longo alcance e algoritmos de visão computacional, associado a uma hierarquia geográfica de áreas. O modelo foi apresentado com mais detalhes através de diagramas UML que ilustram aspectos de implementação desse modelo arquitetural, e um protótipo foi desenvolvido para realizar testes de validação do modelo.

Foram realizados testes nos algoritmos implementados para verificar se eles apresentavam os resultados idealizados na teoria e para prover um bom ambiente de teste de validação do modelo

adaptativo, destacando-se algoritmos para a localização do dispositivo móvel por sinais de Wi-Fi para ambiente interno, para reconhecimento de cena interna/externa e para reconhecimento de imagens para posicionamento correto de marcadores. Os testes apresentaram resultados satisfatórios na grande maioria dos casos, mostrando que é possível desenvolver uma aplicação com essas funcionalidades.

Além disso, os testes de registro de *logs* do posicionamento do GPS e Wi-Fi (para posicionamento interno) mostrou que o posicionamento do GPS possui baixa precisão ao longo de muitas utilizações, tem problemas de baixa acurácia quando o usuário está próximo do POI alvo, que o posicionamento em ambiente interno utilizando GPS é inviável dada as restrições da tecnologia atual, e que adaptar o posicionamento utilizando os sinais de Aps Wi-Fi quando o usuário entra em um prédio soluciona o problema, apesar de ainda apresentar algumas falhas, como a demora na atualização.

Os testes de validação realizados com avaliadores experientes em tecnologia da informação utilizando a aplicação desenvolvida, seguida de entrevista, apontaram que as medidas adaptativas implementadas na aplicação foram tanto percebidas quanto tiveram avaliações positivas, e que a aplicação tem grande potencial para descoberta e exploração de novos lugares, principalmente internos, sendo ideal para setores como o turismo. Como pontos negativos, foram identificados problemas de demora na atualização das informações mostradas aos usuários e confirmou a existência de problemas com a precisão do GPS quando o usuário está próximo ou dentro do POI, fazendo com que os marcadores não sejam desenhados na posição correta.

O uso da visão computacional de forma integrada na aplicação teve avaliações positivas e foi indicado como uma forma inovadora de visualizar POIs ou IOIs e ter uma associação mais precisa entre o marcador virtual e local físico, principalmente quando o usuário está próximo ao POI alvo, uma vez que nessas condições os possíveis erros do sistema de localização são mais perceptíveis.

As adaptações da interface gráfica ajudaram o usuário a se localizar de forma mais adequada em ambos ambientes interno e externo. Uma das adaptações mais evidentes na interface da aplicação foi o mini radar que em ambiente interno mostrou a localização do usuário e dos IOIs na planta baixa do prédio no respectivo andar. Adicionalmente a rota até um POI ou IOI de interesse seguiu um fluxo de passo-a-passo que fez com que os participantes do teste chegassem

ao local desejado independente se este lugar é externo ou interno. Ainda foi sugerido por um dos participantes que a rota até o POI foi ainda mais personalizada com mensagens do tipo: “siga no corredor” e “última porta a direita” para ambientes internos e “atravesse a ponte” e “dobre a direita na próxima esquina” para ambientes externos, o que é possível de ser feito com a adaptação da aplicação.

Considerando o cenário apresentado, esta tese de doutorado contribuiu com o desenvolvimento de um modelo que mitigue os problemas de transição de ambientes internos e externos, de forma que seja possível transitar entre esses ambientes de forma transparente para o usuário. A lista a seguir descreve as contribuições obtidas:

- Melhora na experiência dos usuários no uso de aplicações de RAM:
 - Tornando as aplicações de RAM adaptativas às possíveis mudanças de ambientes interno e externo de forma gradual e transparente para o usuário;
 - Diminuindo o tempo de processamento para calcular o registro com visão computacional utilizando algoritmos conforme a necessidade de acurácia e velocidade de execução.
 - Utilizando um modelo distribuído e hierárquico de apresentação de dados, para não sobrecarregar cognitivamente o usuário, e disponibilizar informações relevantes de acordo com o contexto do mesmo.
- Melhorando o processo de registro combinando a utilização de visão computacional e sensores, buscando diminuir a quantidade de vezes que se utiliza técnicas de reconhecimento de imagem durante a sequência de imagens no vídeo, obtendo um custo benefício em relação à precisão e velocidade;

Como trabalhos futuros têm-se:

- Diminuir a demora em mostrar as informações para o usuário
 - Fazendo uma análise preditiva da caminhada do usuário para saber se o mesmo vai realizar uma transição de ambientes;
 - Realizando estudos com novas tecnologias de rastreamento interno e externo;
 - Realizando testes com outros algoritmos de visão computacional;

- Otimizar os algoritmos envolvidos na localização interna e externa;
 - Inserir o usuário como participante ativo no processo de transição entre ambientes.
- Melhorar a iconografia dos POIs para dar uma melhor sensação de presença do marcador no local do POI;
 - Utilizar técnicas otimizadas de visão computacional para melhor posicionar os marcadores que tem a posição calculada por GPS e sensores.
 - Fazer com que o tamanho e forma dos marcadores variem de acordo com a distância dos POIs dando uma noção de profundidade.
- Desenvolver algoritmos em nuvem para recuperação hierárquica de conteúdos e execução remota de técnicas que demandam alto custo computacional a partir da localização do usuário;
- Desenvolvimento de técnicas de visão computacional baseadas em SLAM para auxiliar o posicionamento do usuário quando houver demora para recuperação da próxima coordenada geográfica ou interna.

REFERÊNCIAS

AHMED, L. et al. **Interaction techniques in mobile Augmented Reality: State-of-the-art**. In: Intelligent Computing and Information Systems (ICICIS), 2015 IEEE Seventh International Conference on. IEEE, pp. 424-433. 2015.

ALCANTARILLA, P.; NUEVO J.; BARTOLI A. **Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces**. Conference: British Machine Vision Conference (BMVC). 2013.

AMERSHI, S. et al. **Power to the people: The role of humans in interactive machine learning**. AI Magazine, vol. 35, nº 4, pp. 105-120, 2014.

AMERSHI, S.; CAKMAK, M.; KNOX, W.; KULESZA, T. **Power to the People: The Role of Humans in Interactive Machine Learning**. AI MAGAZINE. Vol. 35, Nº4, pp. 105-120. 2014.

ANTONACI, A. ; KLEMKE, R. ; SPECHT, M. **Towards Design Patterns for Augmented Reality Serious Games**. Em: International Conference on Mobile and Contextual Learning. Springer International Publishing. Pp. 273-282. Oct. 2015.

AUGMENT. Augment, 2017. Disponível em: <<http://www.augment.com/>>. Acesso em: Fevereiro 2017.

AWILA. AWILA Augmented Technologies, 2017. Disponível em: <<http://www.aug-tech.com/>>. Acesso em: Fevereiro 2017.

AZUMA, R. et al. **Recent Advances in Augmented Reality**. Computer Graphics and Applications, IEEE, Los Alamitos, CA, v. 21, nº 6, pp. 34-47. Novembro 2001.

AZUMA, R. T. **The Most Important Challenge Facing Augmented Reality**. PRESENCE: Teleoperators and Virtual Environments, (00). 2016.

BAI, H. et al. **Markerless 3d gesture-based interaction for handheld augmented reality interfaces**. In: Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on. IEEE, pp. 1-6. 2013.

BAI, H.; GAO, L.; BILLINGHURST, M. **Markerless fingertip-based 3D interaction for handheld augmented reality in a small workspace**. In: 3D User Interfaces (3DUI), 2013 IEEE

Symposium on. IEEE, pp. 129-130. 2013.

BARRY, A. et al. **Augmented reality in a public space: The natural history museum, london.** Computer, vol. 45, n° 7, pp. 42-47, 2012.

BILLINGHURST, M. et al. **A survey of augmented reality.** Foundations and Trends® Human-Computer Interaction, vol. 8, n° 2-3, pp. 73-272, 2015.

BILLINGHURST, M.; KATO, H.; MYOJIN, S. **Advanced interaction techniques for augmented reality applications.** In: International Conference on Virtual and Mixed Reality. Springer Berlin Heidelberg, pp. 13-22. 2009.

BIMBER, O.; RASKAR, R. **Spatial Augmented Reality: Merging Real and Virtual Worlds.** Danvers, MA: A K Peters/CRC Press by Taylor & Francis Group. 2005.

BINU, P. K.; KRISHNAN, R. A.; KUMAR, A. P. **An efficient indoor location tracking and navigation system using simple magnetic map matching.** Em: Computational Intelligence and Computing Research (ICCIC), 2016 IEEE International Conference on. IEEE. Pp. 1-7. 2016.

BLIPPAR. Blippar, 2017. Disponível em: <<https://blippar.com/en/>>. Acesso em: Fevereiro 2017.

BRADSKI, G. **The opencv library.** Doctor Dobbs Journal. Pp. 120-126. 2000.

BRATA, K. C.; LIANG, D.; PRAMONO, S. H. **Location-Based Augmented Reality Information for Bus Route Planning System.** International Journal of Electrical and Computer Engineering, vol. 5, n° 1, pp. 142, 2015.

BREIMAN, L. **“Random forests”** Machine learning. Vol. 45, n° 1, pp. 5-32, 2001.

BUTCHART, B. **Architectural styles for augmented reality in smartphones.** In: Third International AR Standards Meeting. Pp. 1-7. 2011.

CALONDER, M.; et. al. **BRIEF: Computing a Local Binary Descriptor Very Fast.** Pattern Analysis and Machine Intelligence. Vol. 34, N°. 7, pp. 1281-1298. 2012.

CARMO, M. B. et al. **PoI Awareness, Relevance and Aggregation for Augmented Reality.** In: Information Visualisation (IV), 2016 20th International Conference. IEEE, pp. 300-305. 2016.

CARMO, R. M. C. et al. **Coordinated and multiple views in augmented reality environment.** In: Information Visualization, 2007. IV'07. 11th International Conference. IEEE, pp. 156-162. 2007.

CARPENDALE, M. S. T. **Considering visual variables as a basis for information visualisation.** Relatório de Pesquisa, Department of Computer Science, University of Calgary. 2003.

CAUDELL, Thomas P.; MIZELL D. W. **Augmented reality: An application of heads-up display technology to manual manufacturing processes**. Em System Sciences, Proceedings of the Twenty-Fifth Hawaii International Conference on. IEEE. Vol. 2. 1992.

CHANG, Q. et al. **Wi-Fi Fingerprint Positioning Updated by Pedestrian Dead Reckoning for Mobile Phone Indoor Localization**. Proceedings of China Satellite Navigation Conference (CSNC). pp. 729-739. 2015

CHATZOPOULOS, D.; BERMEJO, C.; HUANG, Z.; HUI, P. **Mobile Augmented Reality Survey: From Where We Are to Where We Go**. IEEE Access. 2017

CHEKHLOV, D. et al. **Ninja on a plane: Automatic discovery of physical planes for augmented reality using visual slam**. Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. IEEE Computer Society, pp. 1-4. 2007.

CHEN, C.; CHANG, B. R.; HUANG, P. **Multimedia augmented reality information system for museum guidance**. Personal and ubiquitous computing, vol. 18, n° 2, pp. 315-322, 2014.

CHEN, J.; CAO, R.; WANG, Y. **Sensor-Aware Recognition and Tracking for Wide-Area Augmented Reality on Mobile Phones**. Sensors, vol. 15, n° 12, pp. 31092-31107, 2015.

CHEN, L.; LI, B.; ZHAO, K.; RIZOS, C.; ZHENG, Z.. **An Improved Algorithm to Generate a Wi-Fi Fingerprint Database for Indoor Positioning**. Sensors, vol. 13, n° 8, pp. 11085-11096, 2013.

CHI, H. L.; KANG, S. C.; WANG, X. **Research trends and opportunities of augmented reality applications in architecture, engineering, and construction**. Automation in construction, 33, 116-122. 2013;

CHO, K. et al. **Real-time recognition and tracking for augmented reality books**. Computer Animation and Virtual Worlds, vol. 22, n° 6, pp. 529-541, 2011.

CHOI, J. et al. **Bare-hand-based augmented reality interface on mobile phone**. In: Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on. IEEE, pp. 275-276. 2011.

CHOI, S. **Random Sample Consensus: a Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography**. Communications of the ACM, Vol. 24, N° 6, pp. 381-395. 1981.

CHOWDHURY, S. A. et al. **Handheld augmented reality interaction technique**. In: International Visual Informatics Conference. Springer International Publishing, pp. 418-426. 2013.

CHUN, We. H.; HÖLLERER, T. **Real-time hand interaction for augmented reality on mobile phones**. In: Proceedings of the 2013 international conference on Intelligent user interfaces. ACM, pp. 307-314. 2013.

DE SÁ, M.; CHURCHILL, E. F. **Mobile Augmented Reality: A Design Perspective. Human Factors in Augmented Reality Environments.** New York, NY: Springer, Cap. 6, pp. 139-164. 2013.

DEMENTHON, Daniel F.; DAVIS, Larry S. **Model-based object pose in 25 lines of code.** International journal of computer vision, vol. 15, n° 1-2, pp. 123-141. 1995.

DONAHUE, M. S. **The Field Play Mobile Application: Augmented Reality, Location-Aware Content, And The Creation Of An Interactive, Data-Rich Digital Educational And Research Tool.** Episodic Uplift Of The Rocky Mountains: Evidence From U-Pb Detrital Zircon Geochronology And Low-Temperature Thermochronology With A Chapter On Using Mobile Technology For Geoscience Education, 168. 2016.

DONG, S. et al. **Collaborative visualization of engineering processes using tabletop augmented reality.** Advances in Engineering Software, vol. 55, pp. 45-55, 2013.

DUBROFSKY, E. **Homography Estimation.** The University of British Columbia. Dissertação de Mestrado. Março, 2009.

DÜNSER, A. et al. **Applying HCI Principles in AR Systems Design.** Proceedings of the IEEE Virtual Reality Conference: Mixed Reality User Interfaces: Specification, Authoring, Adaptation Workshop (MRUI'07). Charlotte, NC: IEEE, pp. 37-42. 2007.

DÜNSER, A. et al. **Exploring the use of handheld AR for outdoor navigation.** Computers & Graphics, Elmsford, NY, v. 36, n° 8, pp. 1084-1095. Dezembro 2012.

DURAI SWAMI, R. ; ZOTKIN, D. N. **Efficient physics based simulation of spatial audio for virtual and augmented reality.** The Journal of the Acoustical Society of America, 140(4), 2999-3000. 2016.

EADE, E.; DRUMMOND, T. **Scalable monocular SLAM.** Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Volume 1. IEEE Computer Society, pp. 469-476. 2006.

ENDERT, A. et al. **The human is the loop: new directions for visual analytics.** Journal of intelligent information systems, vol. 43, n° 3, pp. 411-435, 2014.

ENDERT, A.; NORTH, C.; CHANG, R.; ZHOU, M. **Toward Usable Interactive Analytics: Coupling Cognition and Computation.** Workshop on Interactive Data Exploration and Analytics. Workshop on Interactive Data Exploration and Analytics. 2014.

FERNÁNDEZ-PALACIOS, J. B. et al. **ARCube—The Augmented Reality Cube for Archaeology.** Archaeometry, vol. 57, n° S1, pp. 250-262, 2015.

FISCHLER, M. A.; BOLLES, R. C. **Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.** Communications of the ACM, vol. 24, n° 6, pp. 381-395, 1981.

FONTEYN, E.; KUIPERS, B.; GROBE, S. **A Description of Think Aloud Method and Protocol Analysis**. Qualitative Health Research, vol. 3, n°4, pp. 430-441, SAGE Publications, 1993.

FUJI, T.; MITSUKURA, Y.; MORIYA, T. **Furniture layout AR application using floor plans based on planar object tracking**. RO-MAN, 2012 IEEE. IEEE, pp. 670-675. 2012.

FURHT, B. **Handbook of augmented reality**. Springer Science & Business Media. 2011.

GAMMA, E.; VLISSIDES, J.; HELM, R.; JOHNSON, R. **Design patterns: Elements of reusable object-oriented software**. Reading: Addison-Wesley, 49(120), 11. 1995.

GEIGER, P. et al. **Location-based mobile augmented reality applications: Challenges, examples, lessons learned**. Conference on Web Information Systems and Technologies (WEBIST), pp. 383-394. 2014.

GROCE, A. et al. **You are the only possible oracle: Effective test selection for end users of interactive machine learning systems**. IEEE Transactions on Software Engineering, vol. 40, n° 3, pp. 307-323, 2014.

GRUBERT, J.; GRASSET, R. **Augmented Reality for Android application development**. Packt Publishing Ltd. 2013.

GRUBERT, J.; LANGLOTZ, T.; GRASSET, R. **Augmented reality browser survey**. Institute for computer graphics and vision, University of Technology Graz, technical report, n° 1101. 2011.

GUO, T.; QIAO, F.; LIU, M.; Xu, A.; LIU, Q.; ZHANG, N. **An improved Kalman filtering method for indoor location**. Em: Computer and Communications (ICCC), 2016 2nd IEEE International Conference on. IEEE. Pp. 1747-1751. October, 2016.

HALIK, Ł. **The analysis of visual variables for use in the cartographic design of point symbols for mobile Augmented Reality applications**. Geodesy and Cartography, vol. 61, n° 1, pp. 19-30, 2012.

HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. **The WEKA Data Mining Software: An Update**. SIGKDD Explorations, 2009, vol. 11, n° 1.

HARTLEY R.; ZISSERMAN A. **Multiple view geometry in computer vision**. Cambridge University Press; 2 edition. March 25, 2004.

HUANG et al. **Active Learning with Oracle Epiphany**. 30th Conference on Neural Information Processing Systems. Barcelona, Espanha. 2016.

HÜRST, W.; VAN WEZEL, C. **Gesture-based interaction via finger tracking for mobile augmented reality**. Multimedia Tools and Applications, vol. 62, n° 1, pp. 233-258, 2013.

HÜRST, W.; VAN WEZEL, C. **Multimodal interaction concepts for mobile augmented reality applications**. In: International Conference on Multimedia Modeling. Springer Berlin Heidelberg, pp. 157-167. 2011.

JIA, Z.; GALLAGHER, A.; CHEN, T. **Cameras and gravity: Estimating planar object orientation**. In: Image Processing (ICIP), 2013 20th IEEE International Conference on. IEEE, pp. 3642-3646. 2013.

JORDAN, M.; MITCHELL, T. **Machine learning: Trends, perspectives, and prospects**. Science. Vol. 349, N° 6245, pp. 255-260. 2015.

KAMRUZZAMAN, M.; ISLAM, T.; PODDAR, S. R. **Accuracy of Handheld GPS comparing with Total Station in Land Use Survey: A Case Study in RUET Campus**. International Journal of Innovation and Applied Studies, vol. 7, n° 1, pp. 343, 2014.

KANG, W.; HAN, Y. **SmartPDR: Smartphone-Based Pedestrian Dead Reckoning for Indoor Localization**. IEEE Sensors Journal. Vol. 15, No. 5, pp. 2906-2916. 2015.

KANG, J. et al. **Augmented EKF based SLAM method for improving the accuracy of the feature map**. Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on. IEEE, pp. 3725-3731. 2010.

KATO, H.; KATO, T. **A marker-less augmented reality based on fast fingertip detection for smart phones**. In: Consumer Electronics (ICCE), 2011 IEEE International Conference on. IEEE, pp. 127-128. 2011.

KEIM, D.; Munzner, T.; Rossi, F.; Verleysen, M. **Bridging Information Visualization with Machine Learning**. Dagstuhl Reports. Vol. 5, N°3, pp. 1-27. 2015.

KHAN, A.; KHUSROPARKISTAN, S. **The Rise of Augmented Reality Browsers: Trends, Challenges and Opportunities**. Journal of Science. Vol. 67 n° 3. September, 2015.

KIM, D. et al. **Development and Evaluation of Mobile Tour Guide Using Wearable and Hand-Held Devices**. International Conference on Human-Computer Interaction. Springer International Publishing, pp. 285-296. 2016.

KIM, K.; PARK, N.; WOO, W. **Vision-based all-in-one solution for augmented reality and its storytelling applications**. The Visual Computer, vol. 30, n° 4, pp. 417-429, 2014.

KIM, W.; PARK, J.; KIM, C. **A novel method for efficient indoor-outdoor image classification**. Journal of Signal Processing Systems, 61(3), 251-258. 2010.

KLEIN, G.; MURRAY, D. **Parallel tracking and mapping for small AR workspaces**. Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on. IEEE, pp. 225-234. 2007.

KLEIN, G.; MURRAY, D. **Parallel tracking and mapping on a camera phone**. Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on. IEEE, pp. 83-86. 2009.

KNEIP, L.; FURGALE, P.; SIEGWART, R. **Using multi-camera systems in robotics: Efficient solutions to the npnp problem**. Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, pp. 3770-3776. 2013.

KNEIP, L.; LI, H.; SEO, Y. **Upnp: An optimal o (n) solution to the absolute pose problem with universal applicability**. European Conference on Computer Vision. Springer International Publishing, pp. 127-142. 2014.

KNEIP, L.; SCARAMUZZA, D.; SIEGWART, R. **A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation**. Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE. pp. 2969-2976. 2011.

KRIG, S. **Computer Vision Metrics: Survey, Taxonomy, and Analysis**. Apress. 2014.

KURZ, D. et al. **Absolute spatial context-aware visual feature descriptors for outdoor handheld camera localization overcoming visual repetitiveness in urban environments**. In: Computer Vision Theory and Applications (VISAPP), 2014 International Conference on. IEEE, pp. 56-67. 2014.

KURZ, D.; BENHIMANE, S. **Handheld augmented reality involving gravity measurements**. Computers & Graphics, vol. 36, n° 7, pp. 866-883, 2012.

KURZ, D.; HIMANE, S. B. **Inertial sensor-aligned visual feature descriptors**. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, pp. 161-166. 2011.

KURZ, D.; OLSZAMOWSKI, T.; BENHIMANE, S. **Representative feature descriptor sets for robust handheld camera localization**. In: Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on. IEEE, pp. 65-70. 2012.

LANGLOTZ, T. et al. **Next-generation augmented reality browsers: rich, seamless, and adaptive**. Proceedings of the IEEE, vol. 102, n. 2, pp. 155-169, 2014.

LANGLOTZ, T.; GRUBERT, J.; GRASSET, R. **Augmented reality browsers: essential products or only gadgets?**. Magazine Communications of the ACM. Vol. 56 n° 11, pp. 34-36. November 2013.

LAROSE, D. **Discovering knowledge in data: an introduction to data mining**. John Wiley & Sons, 2014.

LAYAR. Layar, 2017. Disponível em: <<https://www.layar.com/>>. Acesso em: Fevereiro 2017.

LEE, G. A. et al. **Antarcticar: An outdoor ar experience of a virtual tour to antarctica**. In: Mixed and Augmented Reality-Arts, Media, and Humanities (ISMAR-AMH), 2013 IEEE

International Symposium on. IEEE, pp. 29-38. 2013.

LEE, G. A. et al. **CityViewAR: A mobile outdoor AR application for city visualization.** In: Mixed and Augmented Reality (ISMAR-AMH), 2012 IEEE International Symposium on. IEEE, p. 57-64. 2012.

LEE, G. A.; BILLINGHURST, M. **A component based framework for mobile outdoor AR applications.** In: Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry. ACM, pp. 207-210. 2013.

LEE, T.; HOLLERER, T. **Handy AR: Markerless inspection of augmented reality objects using fingertip tracking.** In: Wearable Computers, 2007 11th IEEE International Symposium on. IEEE, pp. 83-90. 2007.

LEUTENEGGER, S.; CHLI, M.; SIEGWART, R. **BRISK: Binary robust invariant scalable keypoints.** Int.Conference on Computer Vision (ICCV). 2011.

LI, J. et al. **Camera Pose Estimation by Vision-inertial Sensor Fusion: An Application to Augmented Reality Books.** Electronic Imaging, vol. 2016, n° 4, pp. 1-6, 2016.

LIANG, P. C.; KRAUSE, P. **Smartphone-based real-time indoor location tracking with 1-m precision.** IEEE Journal of Biomedical and Health Informatics, 20(3), 756-762. 2016.

LIAO, T.; HUMPHREYS, L. **Layar-ed places: Using mobile augmented reality to tactically reengage, reproduce, and reappropriate public space.** new media & society, vol. 17, n° 9, pp. 1418-1435, 2015.

MAHRIN, M. N.; STROOPER, P.; CARRINGTON, D. **Selecting Usability Evaluation Methods for Software Process Descriptions.** Proceedings of the 16th Asia-Pacific Software Engineering Conference. Penang: [s.n.], pp. 523-529. 2009.

MARCHAND, E.; UCHIYAMA, H.; SPINDLER, F. **Pose estimation for augmented reality: a hands-on survey.** IEEE transactions on visualization and computer graphics, vol. 22, n° 12, p. 2633-2651, 2016.

MARKOV-VETTER, D. ; STAADT, O. **A pilot study for Augmented Reality supported procedure guidance to operate payload racks on-board the International Space Station.** 2013. IEEE International Symposium on Mixed and Augmented Reality (ISMAR). vol., no., pp.1-6, 1-4. Oct. 2013.

MARTÍNEZ, H. et al. **Drivers and Bottlenecks in the Adoption of Augmented Reality Applications.** Journal on Multimedia Theory and Applications. Vol. 1, pp. 27-44. 2014.

MARTÍNEZ, H.; BANDYOPADHYAY, P. **Analysis of Four Usability Evaluation Methods Applied to Augmented Reality Applications.** Tampere University of Technology e University of Helsinki. Finlândia. 2014.

MARZO, A.; BOSSAVIT, B.; HACHET, M. **Evaluating controls for a point and shoot mobile game: Augmented Reality, Touch and Tilt.** In: Mixed and Augmented Reality-Media, Art, Social Science, Humanities and Design (ISMAR-MASH'D), 2014 IEEE International Symposium on. IEEE, pp. 59-62. 2014.

MASSELLI, A.; ZELL, A. **A New Geometric Approach for Faster Solving the Perspective-Three-Point Problem.** Pattern Recognition (ICPR), 2014 22nd International Conference on. IEEE, p. 2119-2124. 2014.

METAIO. Metaio, 2017. Disponivel em: <<http://www.metaio.eu/>>. Acesso em: Fevereiro 2017.

MILGRAM, P. et al. **Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum.** Telemanipulator and Telepresence Technologies. Boston: [s.n.]. pp. 282-292. 1994.

MIYASHITA, T. et al. **An augmented reality museum guide.** Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. IEEE Computer Society, pp. 103-106. 2008.

MOURAGNON, E. et al. **Real time localization and 3d reconstruction.** Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. IEEE, pp. 363-370. 2006.

MULLONI, A. et al. **Experiences with the impact of tracking technology in mobile augmented reality evaluations.** In: MobileHCI 2012 Workshop MobiVis. ACM, 2012.

NAZRI, N. I. A. M.; RAMBLI, D. R. A. **Current limitations and opportunities in mobile augmented reality applications.** Computer and Information Sciences (ICCOINS), International Conference on. IEEE, pp. 1-4. 2014.

NEALE, S. et al. **Natural interactions between augmented virtual objects.** In: Proceedings of the 23rd Australian Computer-Human Interaction Conference. ACM, pp. 229-232. 2011.

NI, L. M. ; LIU, Y. ; LAU, Y. C.; PATIL, A. P. **LANDMARC: indoor location sensing using active RFID.** Wireless networks, 10(6), 701-710. 2004.

NURMINEN, A.; JÄRVI, J.; LEHTONEN, M. **A mixed reality interface for real time tracked public transportation.** In: 10th ITS European Congress. Helsinki, Finlandiya. 2014.

OLSSON, T.; SALO, M. **Narratives of satisfying and unsatisfying experiences of current mobile augmented reality applications.** In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM, pp. 2779-2788. 2012.

PARK, J. G.; CHARROW, B.; CURTIS, D.; BATTAT, J.; MINKOV, E.; HICKS, J.; ...; LEDLIE, J. **Growing an organic indoor location system.** Em: Proceedings of the 8th international conference on Mobile systems, applications, and services. ACM (pp. 271-284). June, 2010.

PAYNE, A.; SINGH, S. **Indoor vs. outdoor scene classification in digital photographs.**

Pattern Recognition, vol. 38, n° 10, pp. 1533-1545. 2005.

POLLMANN, F. et al. **Evaluation of interaction methods for a real-time augmented reality game.** In: International Conference on Entertainment Computing. Springer Berlin Heidelberg, pp. 120-125. 2013.

POLVI, J. et al. **SlidAR: A 3D positioning method for SLAM-based handheld augmented reality.** Computers & Graphics, vol. 55, pp. 33-43, 2016.

PRAKASH, P.; KANNAMMAL K. **Dynamic Hand Interaction in Handheld AR.** International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, INDIA. Jan. 07 – 09, 2016

PRYSS, R. et al. **Advanced Algorithms for Location-Based Smart Mobile Augmented Reality Applications.** Procedia Computer Science, vol. 94, pp. 97-104, 2016.

PUYUELO, M. et al. **Experiencing Augmented Reality as an Accessibility Resource in the UNESCO Heritage Site Called “La Lonja”, Valencia.** Procedia Computer Science, vol. 25, pp. 171-178, 2013.

RIDDLE, S. ; WASSER, D. ; MCCARTHY, M. **Touching The Human Neuron: User-Centric Augmented Reality Viewing and Interaction of in-vivo Cellular Confocal Laser Scanning Microscopy (CLSM) Utilizing High Resolution zStack Data Sets.** Journal of Biocommunication, 41(1). 2017.

RISH, I. **An empirical study of the naive Bayes classifier.** IJCAI 2001 workshop on empirical methods in artificial intelligence. vol. 3. n°. 22. IBM New York, 2001.

ROSTEN, E.; PORTER, R. ; DRUMMOND, T. **Faster and better: a machine learning approach to corner detection.** IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 32, N°. 1. pp. 105-119. 2010.

ROUSSEEUW, P. **Least Median of Squares Regression.** Journal of the American Statistical Association. Vol. 79, N° 388, pp. 871-880. 1984.

RUBLEE, E. et al. **ORB: An efficient alternative to SIFT and SURF.** International Conference on Computer Vision (ICCV). pp. 2564-2571. IEEE 2011.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach.** 3. ed. New Jersey: Prentice Hall, 2009.

RUTA, M. et al. **Indoor/outdoor mobile navigation via knowledge-based POI discovery in augmented reality.** In: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2015 IEEE/WIC/ACM International Conference on. IEEE, pp. 26-30. 2015.

SAMBINELLI, F.; ARIAS, C. S. **Augmented Reality Browsers: A Proposal for Architectural Standardization.** International Journal of Software Engineering & Applications, vol. 6, n° 1, pp.

1, 2015.

SAMBINELLI, F.; ARIAS, C. S. **Augmented Reality: Applications and Browsers**. International Journal of Advanced Research in Computer Science & Technology, vol. 2, n° 4, pp. 85-89, 2014.

SCHMALSTIEG, D.; WAGNER, D. **Experiences with Handheld Augmented Reality**. Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07). Washington, DC: IEEE Computer Society, pp. 1-13. 2007.

SEIDINGER, I.; GRUBERT, J. **3D Character Customization for Non-Professional Users in Handheld Augmented Reality**. arXiv preprint arXiv:1607.06587, 2016.

SEO, B. et al. **One-handed interaction with augmented virtual objects on mobile devices**. In: Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry. ACM, pp. 8. 2008.

SERRANO, N.; SAVAKIS, A.; LUO, A. **A computationally efficient approach to indoor/outdoor scene classification**. Em Pattern Recognition, Proceedings of 16th International Conference on. IEEE. Vol. 4, pp. 146-149. 2002.

SIMON, D. **Kalman filtering with state constraints: a survey of linear and nonlinear algorithms**. IET Control Theory & Applications, vol. 4, n° 8, pp. 1303-1318, 2010.

SIMON, G.; BERGER, M. O. **Reconstructing while registering: a novel approach for markerless augmented reality**. Mixed and Augmented Reality, 2002. ISMAR 2002. Proceedings. International Symposium on. IEEE, pp. 285-293. 2002.

SIMON, G.; FITZGIBBON, A. W.; ZISSERMAN, A. **Markerless tracking using planar structures in the scene**. Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on. IEEE, pp. 120-128. 2000.

STRASDAT, H.; MONTIEL, J. M. M.; DAVISON, A. J. **Real-time monocular SLAM: Why filter?**. Robotics and Automation (ICRA), 2010 IEEE International Conference on. IEEE, pp. 2657-2664. 2010.

SUTCLIFFE, A. **Designing for user engagement: Aesthetic and attractive user interfaces**. Synthesis lectures on human-centered informatics, vol. 2, n° 1, pp. 1-55, 2010.

SWEENEY, C. et al. **Efficient computation of absolute pose for gravity-aware augmented reality**. In: Mixed and Augmented Reality (ISMAR), 2015 IEEE International Symposium on. IEEE, pp. 19-24. 2015.

SZUMMER, M. ; PICARD, R. W. **Indoor-outdoor image classification. In Content-Based Access of Image and Video Database**. Em: Proceedings., 1998 IEEE International Workshop on (pp. 42-51). IEEE. 1998.

TEEVAN, J. et al. **Understanding the Importance of Location, Time, and People in Mobile Local Search Behavior**. Proceedings of the 13th international conference on human computer interaction with mobile devices and services (MobileHCI '11). New York, NY: ACM, pp. 77–80. 2011.

TELKENAROGLU, C.; CAPIN, T. **Dual-finger 3d interaction techniques for mobile devices**. Personal and ubiquitous computing, vol. 17, n° 7, pp. 1551-1572, 2013.

TIAN, K. et al. **Development of a Multi-viewpoint AR-Based Mobile Learning System for Supporting Lunar Observation**. In: Signal-Image Technology & Internet-Based Systems (SITIS), 2013 International Conference on. IEEE, pp. 1034-1041. 2013.

TIAN, K. et al. **Multi-viewpoint smartphone AR-based learning system for astronomical observation**. International Journal of Computer Theory and Engineering, vol. 6, n° 5, pp. 396-400, 2014.

TILLON, A. B. et al. **A day at the museum: An augmented fine-art exhibit**. Mixed and Augmented Reality-Arts, Media, and Humanities (ISMAR-AMH), 2010 IEEE International Symposium On. IEEE, p. 69-70. 2010.

TRIGGS, B. et al. **Bundle adjustment: a modern synthesis**. Proc of International Workshop on Vision Algorithms: Theory and Practice. 2000.

TUCH, A. N. et al. **The role of visual complexity and prototypicality regarding first impression of websites: Working towards understanding aesthetic judgments**. International Journal of Human-Computer Studies, vol. 70, n° 11, pp. 794-811. 2012.

TULI, N.; MANTRI, A.. **Interaction Techniques: helping user manipulate AR virtual content**. 4th International Conference on Wireless Networks & Embedded Systems (WECON). 2015.

VALBUENA, R., et al. **Accuracy and precision of GPS receivers under forest canopies in a mountainous environment**. Spanish Journal of Agricultural Research 8.4 1047-1057. 2010.

VILLARRUBIA, G.; DE PAZ, J.; DE LA PRIETA, F.; BAJO, J. **Hybrid Indoor Location System for Museum Tourist Routes in Augmented Reality** International Conference on Information Fusion (FUSION). pp. 1-8. 2014.

VINCENT, T.; NIGAY, L.; KURATA, T. **Handheld Augmented Reality: Effect of registration jitter on cursor-based pointing techniques**. In: Proceedings of the 25th Conference on l'Interaction Homme-Machine. ACM, pp. 1. 2013.

VINCENT, T.; NIGAY, L.; KURATA, T. **Precise pointing techniques for handheld augmented reality**. In: IFIP Conference on Human-Computer Interaction. Springer Berlin Heidelberg, pp. 122-139. 2013.

VUFORIA. Vuforia, 2017. Disponível em: <<https://www.vuforia.com/>>. Acesso em: Fevereiro

2017.

WANG, J.; LUO, X. **Purposive Sample Consensus: A Paradigm for Model Fitting with Application to Visual Odometry**. Field and Service Robotics. Springer International Publishing, pp. 335-349. 2015.

WASSERMAN, A. I. **Software Engineering Issues for Mobile Application Development**. Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research. Nova York: ACM, pp. 397-400. 2010.

WHARTON, C.; RIEMAN, J.; LEWIS, C.; POLSON, P. **The Cognitive Walkthrough Method: A Practitioner's Guide**. In: Usability Inspection Methods. New York, NY, USA: John Wiley & Sons, Inc., pp. 105-140. 1994.

WIKITUDE INC. wiktitude See more. Wiktitude SDK, 2017. Disponível em: <<http://www.wiktitude.com/products/wiktitude-sdk/>>. Acesso em: Fevereiro 2017.

WILLIAMS, S.; GREEN, R.; BILLINGHURST, M. **Transform flow: A mobile augmented reality visualisation and evaluation toolkit**. In: Image and Vision Computing New Zealand (IVCNZ), 2013 28th International Conference of. IEEE, pp. 53-58. 2013.

WU, C. **P3.5P: Pose estimation with unknown focal length**. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2440-2448. 2015.

WU, H. K.; LEE, S. W. Y.; CHANG, H. Y. ; LIANG, J. C. **Current status, opportunities and challenges of augmented reality in education**. Computers & Education, 62, 41-49. 2013.

YANG, X. et al. **Vision-inertial hybrid tracking for robust and efficient augmented reality on smartphones**. In: Proceedings of the 23rd ACM international conference on Multimedia. ACM, pp. 1039-1042. 2015.

YOVCHEVA, Z. **User-centred design of smartphone augmented reality in urban tourism context**. Doctoral dissertation, Bournemouth University. 2015

ZÀRUBA, G. V. ; HUBER, M. ; KAMANGAR, F. A. ; CHLAMTAC, I. **Indoor location tracking using RSSI readings from a single Wi-Fi access point**. Wireless networks, 13(2), 221-235. 2007.

ZHENG, Y. et al. **Revisiting the pnp problem: A fast, general and optimal solution**. Proceedings of the IEEE International Conference on Computer Vision, pp. 2344-2351. 2013.

ZHU, Z. et al. **AR-Weapon: live augmented reality based first-person shooting system**. In: Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. IEEE, pp. 618-625. 2015.