

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**MODELO DE APOIO À
NAVEGAÇÃO EM CALÇADAS
PARA PESSOAS COM
DEFICIÊNCIA VISUAL**

MARCELO CABRAL GHILARDI

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul.

Orientador: Prof. Isabel Harb Manssour

**Porto Alegre
2016**

Dados Internacionais de Catalogação na Publicação (CIP)

G424m Ghilardi, Marcelo Cabral

Modelo de apoio à navegação em calçadas para pessoas com deficiência visual / Marcelo Cabral Ghilardi. – 2016.
84 p.

Dissertação (Mestrado) – Faculdade de Informática, Pontifícia Universidade Católica do Rio Grande do Sul.
Orientador: Prof. Isabel Harb Manssour

1. Visão por Computador. 2. Processamento de Imagens.
3. Acessibilidade. 4. Pessoas com Deficiência Visual.
5. Informática. I. Manssour, Isabel Harb. II. Título.

CDD 23 ed. 006.37

Salete Maria Sartori CRB 10/1363
Setor de Tratamento da Informação da BC-PUCRS



Pontifícia Universidade Católica do Rio Grande do Sul
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

TERMO DE APRESENTAÇÃO DE DISSERTAÇÃO DE MESTRADO

Dissertação intitulada "Modelo de Apoio à Navegação em Calçadas para Pessoas com Deficiência Visual" apresentada por Marcelo Cabral Ghilardi como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação, aprovada em 28 de março de 2016 pela Comissão Examinadora:

Isabel Harb Manssour

Profa. Dra; Isabel Harb Manssour –
Orientadora

PPGCC/PUCRS

Márcio Sarroglia Pinho

Prof. Dr. Márcio Sarroglia Pinho –

PPGCC/PUCRS

Edgard Afonso Lamounier Jr.

Prof. Dr. Edgard Afonso Lamounier Jr. -

UFU

Homologada em 21/07/2016, conforme Ata No. 014 pela Comissão Coordenadora.

Luiz Gustavo Leão Fernandes

Prof. Dr. Luiz Gustavo Leão Fernandes
Coordenador.

DEDICATÓRIA

Dedico esta dissertação a minha esposa que amo tanto e que incansavelmente me acompanhou durante toda a trajetória, abdicou de fim de semanas, viagens, saídas, para sempre estar ao meu lado. Além disso, me cuidou durante períodos difíceis que passamos e me deu força e incentivo para avançar nas pesquisas.

“O pessimista vê dificuldade em cada oportunidade; o otimista vê oportunidade em cada dificuldade.”

(Winston Churchill)

AGRADECIMENTOS

À prof. Dra. Isabel Harb Manssour, minha orientadora, pela oportunidade de trabalhar ao seu lado, por sua dedicação, competência e especial atenção aos detalhes, além do indiscutível apoio e compreensão em momentos difíceis.

À empresa Progress Informática, onde trabalho, em especial à Roni Rui Frantz, Renê Frantz e Sérgio Silva, diretores da empresa, que de forma muito tranquila, e sem nenhum impedimento, me apoiaram neste projeto e me deram as condições de necessárias para iniciar e prosseguir no mestrado.

Aos bolsistas de iniciação científica Rafael Cauduro Oliveira Macedo, Lucas Pedreira da Silveira e Daniel Centeno Einloft pela inestimável ajuda no desenvolvimento do modelo e pela parceria no período de testes com usuários.

À Hewlett-Packard Brasil por fornecer a bolsa de estudo para mestrado e a Pontifícia Universidade Católica do Rio Grande do Sul por todos os recursos físicos, humanos e tecnológico que disponibiliza a seus pesquisadores.

Aos professores e colegas de mestrado, e agora amigos, por formarem uma grande família e por sempre estarem disponíveis para ajudar.

Aos meus familiares que sempre acreditaram em mim e a minha mãe que acha que eu sou um gênio. Aos meus grandes amigos por entenderem o motivo das muitas vezes, durante o estudos, que eu não podia estar presente, além disso, pediram este agradecimento.

À minha gata Lilikit ao meu lado em todas as horas de escrita desta dissertação e principalmente a minha esposa, Marisa Villa, que generosamente sempre me apoiou e se dedicou ao mestrado tanto quanto eu.

MODELO DE APOIO À NAVEGAÇÃO EM CALÇADAS PARA PESSOAS COM DEFICIÊNCIA VISUAL

RESUMO

No Brasil, há mais de 35,7 milhões de pessoas com deficiência visual, sendo que destas 6,5 milhões são incapazes enxergar de modo algum ou possuem grande dificuldade mesmo com lentes. A bengala longa e o cão guia são os meios mais utilizados para auxiliar a mobilidade destes indivíduos. Porém, a bengala longa oferece pouca informação e o cão guia é um recurso de difícil acesso. Considerando este contexto, existem vários estudos relacionados ao auxílio a navegação em ambientes urbanos para indivíduos com deficiência visual, porém ainda há muitas lacunas para serem pesquisadas. O objetivo deste trabalho é apresentar um modelo que, baseado em técnicas de visão computacional, possa auxiliar indivíduos com deficiência visual a trafegar em calçadas. Este modelo apresenta uma solução integrada para localização e identificação de piso tátil, detecção obstáculos aéreos, detecção de obstáculos terrestres e localização de faixas de pedestres. As informações do ambiente são obtidas através de vários meios, tais como câmera, sensor de ultrassom e imagens de satélite, e o *feedback* ao usuário é sonoro. Resultados experimentais são apresentados e demonstram a viabilidade desta abordagem.

Palavras-Chave: Visão Computacional, Processamento de Imagens, Acessibilidade, Detecção de Obstáculos, Deficiente Visual, Piso Tátil.

SUPPORT MODEL FOR NAVIGATING SIDEWALKS FOR VISUALLY IMPAIRED PERSONS

ABSTRACT

There are more than 35.7 million people with visual impairments in Brazil, of these 6.5 million are totally blind or have great difficulty to see, even with lenses. The white cane and the guide dog are the most used resources to support the mobility of those individuals. However, the white cane offers little information and the guide dog is a resource difficult to access. Considering this context, there are several studies related to navigation support in urban environments for individuals with visual impairments, but there are still many gaps to be researched. The main goal of this work is to present a model that, based on computer vision techniques, can help people with visual impairments to walk on sidewalks. This model offers an integrated solution for location and identification of tactile paving, detection of aerial obstacles, detection of obstacles on the ground and location of crosswalks. The environment information is obtained in several ways, such as camera, ultrasonic sensor and satellite images, and the feedback to the user is sonorous. Experimental results are presented and demonstrate the viability of this approach.

Keywords: Computer Vision, Image Processing, Accessibility, Obstacle Detection, Visually Impaired, Tactile Paving .

LISTA DE FIGURAS

Figura 2.1 – Conjunto de etapas padrão identificadas nos modelos apresentados	32
Figura 3.1 – Modelo geral proposto	36
Figura 3.2 – Exemplo de pisos táteis: (a) Piso Direcional; (b) Piso de Alerta	37
Figura 3.3 – Modelos de cores de pisos táteis	37
Figura 3.4 – Exemplos de obstáculos em calçadas: (a) Canteiro; (b) Esquina; (c) Parede	38
Figura 3.5 – (a) Imagem de satélite; (b) imagem rodoviária	40
Figura 4.1 – Ilustração da aquisição da imagem	44
Figura 4.2 – Abordagem para a detecção de piso tátil	45
Figura 4.3 – Histogramas de pisos táteis: (a) tipo A; (b) tipo B	46
Figura 4.4 – Exemplo da aplicação dos limiares e operadores morfológicos: (a) Imagem Original; (b) Após limiares; (c) Após operações morfológicas	46
Figura 4.5 – Exemplo de área detectada: (a) Original; (b) Tipo A; (c) Tipo B	47
Figura 4.6 – Processo de detecção de linhas	48
Figura 4.7 – Comparação entre diferentes detectores de bordas	49
Figura 4.8 – Bloco Segmentado	49
Figura 4.9 – Exemplo de pisos táteis com ruídos	50
Figura 4.10 – Abordagem para a detecção de obstáculos terrestres	51
Figura 4.11 – Exemplo de imagem com obstáculos: (a) Imagem dividida em blocos; (b) Características de cores de cada bloco	52
Figura 4.12 – Blocos da imagem que são processados	53
Figura 4.13 – Detecção de obstáculos aéreos do modelo proposto	54
Figura 4.14 – Equipamento para detecção de obstáculos aéreos	55
Figura 4.15 – Localização de faixas de pedestres e esquinas do modelo proposto	56
Figura 4.16 – A aquisição das imagens: (a) Coordenadas GPS enviadas pelo usuário, ilustrada pelo ponto vermelho; (b) Imagem de satélite extraída do Google Maps; (c) imagem rodoviária extraída do mesmo conjunto de dados	57
Figura 4.17 – Segmentação da imagem: (a) Segmentação inicial na imagem rodoviária; (b) Saída após operações morfológicas - Imagem binária; (c) Junção da imagem de satélite com as regiões de interesse	58
Figura 4.18 – Exemplo da extração manual de amostras (a) positivas e (b) negativas	59

Figura 4.19 – Detecção de faixas de pedestres: (a) Região recortada de uma imagem em análise com um bloco de referência 30×30 marcado por um quadrado vermelho; (b) Bloco de referência e sua célula de referência 15×15 destacada por um quadrado verde; (c) Células de referência utilizadas na segunda verificação; (d) Imagem de entrada satélite (com a região recortada ilustrada na imagem (a) em destaque); (e) <i>Ground truth</i> informado manualmente por um usuário; (f) Estimativa final, dado pelo modelo proposto.	60
Figura 4.20 – Ilustração da linha da rua, esquinas e faixas de pedestres ligadas a quadra onde o usuário está.	61
Figura 4.21 – Exemplo de possíveis <i>feedbacks</i>	62
Figura 5.1 – Exemplo de <i>ground truth</i> para pisos táteis: (a) imagem original; (b) imagem do <i>ground truth</i>	66
Figura 5.2 – Imagens dos testes com os voluntários: (a) apenas com o uso da bengala; (b) "vestido" com o equipamento; (c) com uma pessoa carregando parte do equipamento.	68
Figura 5.3 – Avaliação de acurácia do método GLCM.	71
Figura 5.4 – Ilustração da imagem de satélite em (a) e um caso de faixa mal pintada (ampliado para fins de visualização) em (d) seus respectivos <i>ground truth</i> em (b) e (e). As imagens (c) e (f) ilustram as respectivas saídas de imagens binárias de o modelo proposto para a detecção de faixas.	72

LISTA DE TABELAS

Tabela 2.1 – Resumo dos trabalhos relacionados ordenados por objetivo de detecção e <i>feedback</i>	28
Tabela 2.2 – Resumo dos trabalhos relacionados por tempo de processamento, acurácia e se é vestível	31
Tabela 4.1 – Fórmula para formação de limiares.	47
Tabela 4.2 – Exemplo de <i>feedbacks</i> , ilustrados na figura 4.21.	63
Tabela 5.1 – Resultados da Avaliação	66
Tabela 5.2 – Avaliação de detecção de pisos táteis	67
Tabela 5.3 – Respostas dos testes dos voluntários	68
Tabela 5.4 – Métodos para extração de características avaliadas.	70
Tabela 5.5 – Comparação entre métodos de extração de características.	71
Tabela 5.6 – Avaliação de detecção de faixas.	72
Tabela 5.7 – Avaliação de localização de faixas.	73

LISTA DE SIGLAS

ADVBG – Associação dos Deficientes Visuais de Bento Gonçalves
AM – Aprendizado de Máquina
API – Application Programming Interface
BWS – Black & White Symmetry
DD – Degree of Direction
ETA – Electronic Travel Aids
FN – False Negative
FP – False Positive
GFB – Governo Federal do Brasil
GLCM – Gray-Level Co-Occurrence Matrix
GS – Geometric Symmetry
HSI – Espaço de cor Hue, Saturation e Intensity
HSV – Espaço de cor Hue, Saturation e Value
HT – Transformada de Hough
IBGE – Instituto Brasileiro de Geografia e Estatística
LBP – Local binary patterns
ONU – Organização das Nações Unidas
PDA – Personal Digital Assistant
RFID – Radio-Frequency IDentification
RGB – Espaço de cor Red, Green e Blue
RGB-D – Espaço de cor Red, Green, Blue e Deep
RNA – Redes Neurais Artificiais
SVM – Support Vector Machines
TN – True Negative
TP – True Positive
UFRN – Universidade Federal do Rio Grande do Norte

SUMÁRIO

1	INTRODUÇÃO	23
2	TRABALHOS RELACIONADOS	27
2.1	DETECÇÃO DE FAIXAS DE PEDESTRES	27
2.2	DETECÇÃO DE PISO TÁTIL	29
2.3	AUXÍLIO À NAVEGAÇÃO INTERIOR	29
2.4	DETECÇÃO DE OBSTÁCULOS	30
2.5	ANÁLISE DOS TRABALHOS	31
3	MODELO DE APOIO À NAVEGAÇÃO EM CALÇADAS	35
3.1	METODOLOGIA	35
3.2	DESCRIÇÃO DO MODELO	36
4	IMPLEMENTAÇÃO DO MODELO	43
4.1	ENTRADA DE DADOS	43
4.2	PISOS TÁTEIS	44
4.3	OBSTÁCULOS TERRESTRES EM CALÇADAS	51
4.4	OBSTÁCULOS AÉREOS	53
4.5	FAIXAS DE PEDESTRES E ESQUINAS	56
5	RESULTADOS OBTIDOS	65
5.1	DETECÇÃO DE PISOS TÁTEIS E OBSTÁCULOS	65
5.1.1	AVALIAÇÃO DO ALGORITMO DE PISOS TÁTEIS	65
5.1.2	TESTES COM VOLUNTÁRIOS	67
5.2	DETECÇÃO E LOCALIZAÇÃO DE FAIXAS DE PEDESTRES	69
6	CONCLUSÃO	75
	REFERÊNCIAS	77
	ANEXO A – Termo de consentimento	81
	ANEXO B – Questionário pós-teste	83

1. INTRODUÇÃO

No mundo há mais de 160 milhões de pessoas com algum tipo de deficiência visual, entre estas pessoas, 37 milhões são incapazes de enxergar (totalmente cegas) [19]. No Brasil, de acordo com o Censo Demográfico do Instituto Brasileiro de Geografia e Estatística (IBGE) de 2010 [20], há mais de 35,7 milhões de brasileiros com deficiência visual, destes 6,5 milhões não conseguem enxergar de modo algum ou possuem grande dificuldade mesmo com lentes.

No Brasil, a questão da acessibilidade para pessoas com deficiências sensoriais ou com mobilidade reduzida está estabelecida pela Lei nº 10.098/2000 e regulamentada pelo Decreto Federal do Brasil nº 5296/2004 [16]. Em 2009, a convenção da Organização das Nações Unidas (ONU) sobre direitos das pessoas com deficiência foi promulgada e incorporada à legislação brasileira [17]. Contudo, ainda é possível observar dificuldades para que esse público tenha seus direitos assegurados e respeitados.

Atualmente, embora existam muitos estudos de sistemas de navegação e ferramentas de auxílio à mobilidade para indivíduos com deficiência visual, em nosso país a bengala longa e cão guia ainda são os meios de auxílio mais utilizados. Porém, o cão guia é um recurso inadequado à nossa atual realidade socioeconômica e cultural [31]. O motivo é a complexidade da seleção, aquisição, treinamento e manutenção do animal, aliada a poucas escolas e treinadores no Brasil. Também deve-se considerar a efemeridade da vida do cão e a necessidade de legislação especial para permitir seu uso em qualquer situação. Além disso, muitas pessoas com deficiência visual são idosos e podem ter dificuldade para cuidar adequadamente de animais [41].

A bengala longa é o recurso mais simples, mais barato e, por isto, o auxílio à navegação mais popular. No entanto, a bengala longa não fornece todas as informações necessárias, tais como descrição dos obstáculos, a presença de obstáculos aéreos, distâncias e pontos de referências, entre outras informações que normalmente são identificadas com a visão e que são necessárias para a percepção e o controle em uma locomoção [15].

Barlow e Bond [3] realizaram um estudo que também evidenciou a dificuldade que pessoas com deficiência visual têm ao se locomover em áreas desconhecidas. Por exemplo, as tarefas envolvidas para atravessar uma rua incluem a sua detecção, a localização da faixa de pedestres ou o local para cruzar, a verificação da existência de um semáforo, a determinação do tempo apropriado para cruzar, e caminhar em uma linha reta ao atravessar.

Estudos relacionados ao auxílio a navegação em ambientes urbanos para pessoas com deficiência visual já foram propostos. Por exemplo, métodos como os que auxiliam a travessia de faixas de pedestres [44, 34, 23], a desviar de obstáculos [42, 37, 32], a seguir pisos táteis [26, 24], e a detectar escadas [44], apresentam resultados individuais pro-

missões e possibilidades de melhorias e integrações que motivam pesquisas nesta área. Entretanto, algumas carências ainda presentes nestes métodos criam uma lacuna entre o que é proposto cientificamente e o que pode ser usado na prática.

A navegação em calçadas apresenta muitas dificuldades, tais como: o desvio de obstáculos terrestres e aéreos; a detecção de pessoas e animais; a percepção de cordão de calçada; a localização de faixas de pedestres em ruas e esquinas; ou apenas saber se está sobre uma calçada.

Neste contexto, então, surge a seguinte pergunta de pesquisa: *É possível projetar e desenvolver um novo modelo de apoio à navegação autônoma em calçadas, baseado em visão computacional, que auxilie pessoas com deficiência visual ?*

Portanto, o objetivo geral deste trabalho, considerando esta pergunta de pesquisa, consiste em investigar, projetar e desenvolver algoritmos para comporem um novo modelo para auxiliar pessoas com deficiência visual na navegação autônoma em calçadas. Como resultado da pesquisa realizada, foi desenvolvido um modelo baseado em técnicas de visão computacional, que é composto por quatro partes:

- Detecção de piso tátil;
- Detecção de obstáculos e buracos em calçadas;
- Detecção de obstáculos aéreos;
- Localização de faixas de pedestres;

Um sensor ultrassônico é usado para detecção de obstáculos aéreos. Nas demais partes são usadas técnicas de visão computacional em imagens. Para a detecção de piso tátil e obstáculos em calçadas são utilizadas imagens obtidas através de uma câmera, e para a localização de faixas de pedestres se utilizam imagens de satélite obtidas através de uma biblioteca pública.

Espera-se que este modelo possa ter uma contribuição social, quando aplicado em um contexto global. Isto significa que não se pretende criar uma única e definitiva solução para as dificuldades na locomoção de pessoas com deficiência visual, mas fornecer um incremento à percepção do ambiente que auxilie a locomoção.

Para isto, foram feitas algumas avaliações iniciais do modelo. A localização das faixas de segurança foi avaliada através do uso de um *ground truth* e coordenadas de GPS simuladas. Um *ground truth* também foi gerado e usado para a avaliação da detecção do piso tátil. Posteriormente, alguns testes com voluntários foram feitos para detecção de piso tátil e obstáculos.

Os resultados preliminares mostram que as principais contribuições do modelo proposto são: (1) combinação de imagens de baixa resolução de satélite com sistema

GPS para ajudar as pessoas com deficiência visual na localização de faixas de pedestres; (2) boas taxas de acurácia para detecção (96,9% de acurácia) e localização (92,7% de acurácia) de faixas de pedestre e também para identificação do piso tátil (88,48% de acurácia); (3) detecção de diversos padrões de cores de pisos táteis utilizados no Brasil e diferenciando os pisos direcionais dos pisos de alerta; (4) possibilidade de detectar obstáculos na calçada analisando apenas os padrões de cores.

O restante do documento está organizado da seguinte maneira: O próximo capítulo apresenta uma breve análise de trabalhos relacionados a modelos de auxílio à navegação para pessoas com deficiência visual. O modelo proposto e sua implementação estão descritos, respectivamente, no terceiro e no quarto capítulo. Na sequência, é apresentada uma avaliação do modelo e testes com voluntários. O último capítulo contém as conclusões e os trabalhos futuros.

2. TRABALHOS RELACIONADOS

Este capítulo apresenta uma análise de trabalhos apresentados na literatura com o objetivo de auxiliar o deslocamento de pessoas com deficiência visual. Esta análise foi realizada considerando os objetivos e as técnicas desenvolvidas para alcançá-los, o hardware utilizado, o desempenho, a acurácia e o *feedback* fornecido em cada um dos trabalhos.

Em geral, os trabalhos descritos se destacam pelas técnicas utilizadas para alcançar o objetivo proposto e/ou pelo seu *feedback*, ou seja, apresentam um novo método e/ou uma nova forma de *feedback* para o usuário.

Os *feedbacks* apresentados pelos trabalhos pesquisados e descritos nesta seção se dividem em duas categorias: sonoro ou tátil. A categoria sonoro, por sua vez, pode ser dividida em dois outros tipos: sinais sonoros (como bips) e verbal (voz sintetizada ou gravada). Os trabalhos na categoria tátil se diferem pelo local do corpo no qual o usuário recebe o *feedback*, como mãos, braços, tronco, entre outros, e pelo tipo de sensação tátil, que pode ser desde uma vibração até pequenos estímulos elétricos.

A tabela 2.1 apresenta os trabalhos pesquisados destacando seus objetivos principais, ou seja, o que eles visam identificar, e a forma de *feedback* disponibilizada ao usuário. Nas seções a seguir estes trabalhos são brevemente descritos, agrupados conforme seus objetivos.

2.1 Detecção de faixas de pedestres

Os trabalhos apresentados por Ivanchenko et al. [23], Wang e Tian [44] e Sousa e Marengoni [39], que visam detectar faixas de pedestres, resolvem parcialmente o problema proposto. Eles apresentam bons resultados na detecção da faixa quando o usuário está localizado em frente a faixa, todavia, mesmo que o usuário esteja há poucos metros da faixa, estes métodos não identificam a faixa de pedestres. O trabalho de Wang e Tian [44] também tem o objetivo de detectar escadas, porém, não funciona em ambientes que tenham a luz direta do Sol, isto é, só é possível detectar escadas se for em um local sem a incidência da luz solar.

Para capturar informações do ambiente, os modelos de Ivanchenko et al. [23] e Sousa e Marengoni [39] são compostos apenas por um *smartphone* equipado com câmera. O modelo de Wang e Tian [44] utiliza uma câmera RGB-D que fornece um mapa de profundidade da cena capturada.

Nestes trabalhos são apresentados vários métodos para a identificação da faixa de pedestres. O primeiro método utiliza transformada de Hough para detectar as linhas e um classificador SVM para identificá-las como faixas [44]. O segundo e o terceiro utilizam,

Tabela 2.1 – Resumo dos trabalhos relacionados ordenados por objetivo de detecção e *feedback*

Autor	Ano	Objetivo de detecção	Feedback
Ivanchenko et al. [23]	2008	Faixa de pedestre	Sonoro
Wang e Tian [44]	2012	Faixa de pedestres	Verbal
Sousa e Marengoni [39]	2012	Faixa de pedestre	Verbal
Jie et al. [24]	2010	Piso tátil	Sonoro
Kumar e Sibü [27]	2015	Piso tátil	Verbal
Fallah et al. [14]	2012	Navegação interior	Verbal
Hub et al. [18]	2003	Navegação interior	Verbal
Simoës e Lucena [38]	2016	Navegação interior	Verbal
Aguerrevere et al. [1]	2004	Obstáculos	Sonoro 3D
Ricardo Valentim [42]	2014	Obstáculos	Verbal
Shoval et al. [37]	1994	Obstáculos	Sonoro - osso condução
Tudor et al. [40]	2015	Identificação de objetos	Sonoro
Bousbia-Salah et al. [6]	2007	Obstáculos	Tátil - nos braços
Johnson e Higgins [25]	2006	Obstáculos	Tátil - cinto de vibração
Christian Lykawka [30]	2013	Obstáculos	Tátil - matriz abdômen 7x5
Bourbakis et al. [4]	2013	Obstáculos	Tátil - colete de vibração
Bouzit et al. [7]	2006	Obstáculos	Tátil - dedos da mão
Meers e Ward [32]	2005	Obstáculos	Tátil - mãos
Ito et al. [22]	2005	Obstáculos	Tátil - tensão elétrica
Vera et al. [43]	2014	Obstáculos	Tátil - bengala virtual

respectivamente, um algoritmo detector de bordas customizado [23] e um algoritmo detector de contornos para encontrar as linhas, analisá-las e defini-las como faixa de pedestre [39].

Quando detecta a faixa de pedestres, em um dos trabalhos o *feedback* para o usuário é feito através de um aviso sonoro [23] (beep). Os demais [39] [44] fornecem *feedback* de forma verbal, com informações como distância e segurança.

2.2 Detecção de piso tátil

O formato e cor dos pisos táteis não são padronizados, cada país ou região definem seu próprio padrão. Por este motivo, apesar de terem o mesmo objetivo, os trabalhos tem aplicabilidade em localidade específica.

O trabalho de Jie et al. [24] detecta piso tátil apenas de cor amarela, de um padrão utilizado em uma região da China. Todavia, no Brasil existem pisos táteis com várias cores, como azul, cinza e outras. O trabalho de Woo et al. [45, 46] detecta pisos táteis de padrão específico da Coreia do Sul, que são diferentes em cor e no formato dos utilizados no Brasil.

Nestes trabalhos, as imagens podem ser capturadas do ambiente através de um PDA em tempo real [24] ou são apenas utilizadas imagens armazenadas com o intuito de validar o modelo [45, 46].

Para a detecção de pisos táteis na imagem, a estratégia base adotada pelos modelos [24, 45, 46] é a identificação do piso por segmentação de cor (HSV e HSI, respectivamente) e algoritmo de detecção de bordas (*kirsch* e *Canny*, respectivamente). Entretanto, a análise dos resultados da segmentação de cor e detecção de bordas são bem diferentes nos modelos.

O trabalho de Jie et al. [24] fornece um *feedback* verbal sobre a existência ou não de um piso tátil. Os demais trabalhos [45, 46] fornecem apenas uma tabela de resultados de acurácia e performance.

Uma abordagem diferente é apresentada nos trabalhos de Kumar e Sibin [27] e Kassim et al. [26], que propõe inserir tags RFID sob todos os pisos táteis ou outros pisos com o intuito de, além de detectar pisos táteis, fornecer informações de localização e do ambiente circundante. Foi desenvolvido um protótipo, adaptado em uma bengala longa, com leitor de tags RFID e outros componentes *off-the-shelf*, como *Arduino*, *smartphone* e sensores. O protótipo detecta tags RFID e verifica em uma base de dados se há informações sobre o ambiente circundante relacionadas às tags para fornecer um *feedback* verbal ao usuário. Este tipo de proposta depende de regulamentação e outros entraves burocráticos, além de ter um custo mais alto, todavia, países como China e Japão [27, 26, 10] já desenvolvem sistemas de navegação para cegos utilizando esta tecnologia.

2.3 Auxílio à navegação interior

Alguns trabalhos possuem o objetivo de auxiliar o deslocamento e navegação no interior de prédios e casas. Identificação de escadas, aberturas e móveis são exemplos de questões a serem resolvidas neste caso.

O trabalho de Fallah et al. [14], por exemplo, propõe guiar um usuário cego em um ambiente fechado. Para isso, utiliza informações de deslocamento do usuário obtidas através de um sensor acelerômetro de um *smartphone* e confirmações do usuário quanto a pontos conhecidos do ambiente, como portas, paredes e corredores. O sistema fornece um *feedback* verbal redirecionando o usuário ao seu destino.

Já o trabalho de Hub et al. [18] propõe fornecer informações sobre objetos presentes no ambiente. Sendo assim, foi implementado um protótipo que possui câmera estéreo, bússola digital, sensor de inclinação e teclado. Através dos componentes do protótipo e dados armazenados de um modelo 3D do ambiente, fornece, através de *feedback* verbal, informações como características, cor e distância dos objetos.

O trabalho de Simoes e Lucena [38] apresenta um sistema vestível de navegação interior com base no reconhecimento marcadores visual e detecção de obstáculos por sensor ultrassônico e outros sensores. O *feedback* por mensagens sonoras. As informações são obtidas em tempo real. Dez usuários cegos avaliaram. Os resultados mostraram taxas de cerca de 94,92% para reconhecimento de marcadores e 98,33% de detecção de obstáculos.

Os trabalhos [14, 18, 38] apresentam bons resultados, porém, ambos dependem de um prévio conhecimento sobre o local onde ocorrerá a navegação.

2.4 Detecção de obstáculos

Para o objetivo de detectar obstáculos já existem boas soluções. A contribuição destes trabalhos é, em geral, a apresentação de novos métodos de *feedback* para o usuário. A tabela 2.1 apresenta uma lista dos trabalhos e os métodos de *feedbacks*.

Para detectar objetos e suas distâncias, muitos trabalhos utilizam sensores ultrassônicos. Estes sensores são de baixo custo e de fácil implementação, e, em geral, fornecem a distância de objetos com precisão de centímetros. Shoval et al. [37], Aguerrevere et al. [1], Bouzit et al. [7], Bousbia-Salah et al. [6], Ito et al. [22] e Ricardo Valentim [42] são autores que apresentam alguns exemplos de trabalhos para detecção de obstáculos e que diferem-se, principalmente, pelo número e ângulo dos sensores utilizados, e pela forma de *feedback* apresentada.

Outra forma comumente apresentada para detectar objetos é a utilização de câmeras estéreo ou câmeras com sensor de profundidade. Os trabalhos de Johnson e Higgin [25], Bourbakis et al. [4], Meers e Ward [32] e Christian Lykawka [30] são exemplos que utilizam este recurso. O trabalho de Vera et al. [43], também utiliza uma câmera, mas, neste caso, foi implementado um sensor de profundidade próprio, utilizando feixe de laser em

seu protótipo. A vantagem do uso de câmera é que, além da informação de distância dos objetos, permite obter outras informações como, por exemplo, cor e formato dos objetos.

2.5 Análise dos Trabalhos

Uma forma de definir a usabilidade do modelo é considerar três fatores: seu tempo de processamento, sua acurácia e se seu protótipo é vestível ou como é usado. A tabela 2.2 apresenta o resumo de trabalhos com informações que consideram estes fatores.

Todas as informações referentes a tempo de processamento, acurácia e se é vestível foram extraídas dos trabalhos originais. Em relação ao tempo de processamento não está sendo considerada a diferença entre os equipamentos utilizados nos testes. Quanto à acurácia, não foram consideradas as diferenças nas fórmulas de cálculo de cada trabalho e a definição de vestível considera apenas o fato do usuário poder vestir o equipamento e ficar com as mãos livres, sem considerar o peso e o tamanho destes equipamentos.

Tabela 2.2 – Resumo dos trabalhos relacionados por tempo de processamento, acurácia e se é vestível

Ref.	Objetivo de detecção	Tempo Proc.	Acurácia	Vestível
[23]	Faixa de pedestre	Sim, 3 fps	72%	Segura na mão
[39]	Faixa de pedestres	Sim	76%	Segura na mão
[44]	Faixa de pedestres/escadas	Sim	94%	Não informado
[24]	Piso tátil	Sim, 2 fps	Satisfatório	Segura na mão
[27]	Piso tátil	Sim	100% a 25 pol.	Instalado na bengala ou sapato
[14]	Navegação interior	Sim	85%	Segura na mão
[18]	Navegação interior	Quase	Não informado	Segura na mão
[38]	Navegação interior	Sim	94,92%	Sim
[40]	Identificação de objetos	Sim	Satisfatório	Sim
[1]	Obstáculos	Lento	84, 7% e 99, 2%	Sim
[42]	Obstáculos	Sim	Não informado	Sim
[37]	Obstáculos	Sim	Satisfatório	Sim
[6]	Obstáculos	Sim	Satisfatório	Sim
[25]	Obstáculos	Sim, 10 fps	Satisfatório	Sim
[4]	Obstáculos	Sim, 3 fps	80% a 92%	Sim
[7]	Obstáculos	Sim	52% a 58%	Segura na mão
[32]	Obstáculos	Sim, 15 fps	Satisfatório	Sim
[22]	Obstáculos	Sim	90%	Sim
[43]	Obstáculos	Sim, 13 fps	58%	Segura na mão
[30]	Obstáculos	Sim	Taxas de acerto positivas	Sim

Por se tratar de auxílio a mobilidade, os protótipos desenvolvidos devem processar suas informações em tempo real, isto é, devem fornecer *feedback* instantâneo ao usuário. Entre os modelos apresentados, apenas três, Hub et al. [18], Sainarayanan et al. [35], Aguerrevere et al. [1] não funcionam em tempo real.

Na tabela 2.2 a coluna de acurácia foi preenchida conforme informações dos artigos. Porém, como os modelos possuem objetivos diferentes, não existe uma única medida

de acurácia possível para todos eles. No caso dos protótipos com baixa acurácia, os próprios autores informam que seus modelos precisam ser aprimorados, além disso, alguns não apresentaram dados quantitativos de sua acurácia ou apenas informaram a acurácia como satisfatória.

Neste trabalho, considera-se que um modelo está na categoria de tecnologia vestível, quando o usuário pode vestir o protótipo, seja um colete, uma pulseira ou um boné, sem precisar segurar o protótipo na mão. Entre os 19 trabalhos relacionados, 10 são vestíveis, o que demonstra a relevância de deixar o usuário com suas mãos livres durante sua navegação.

A partir do estudo realizado e da análise destes trabalhos, foi possível identificar um conjunto de etapas padrão para estes modelos. Conforme ilustra a figura 2.1, inicialmente é feita a captura das informações do ambiente, para posterior processamento dessas informações e fornecimento ao usuário do resultado do processamento.

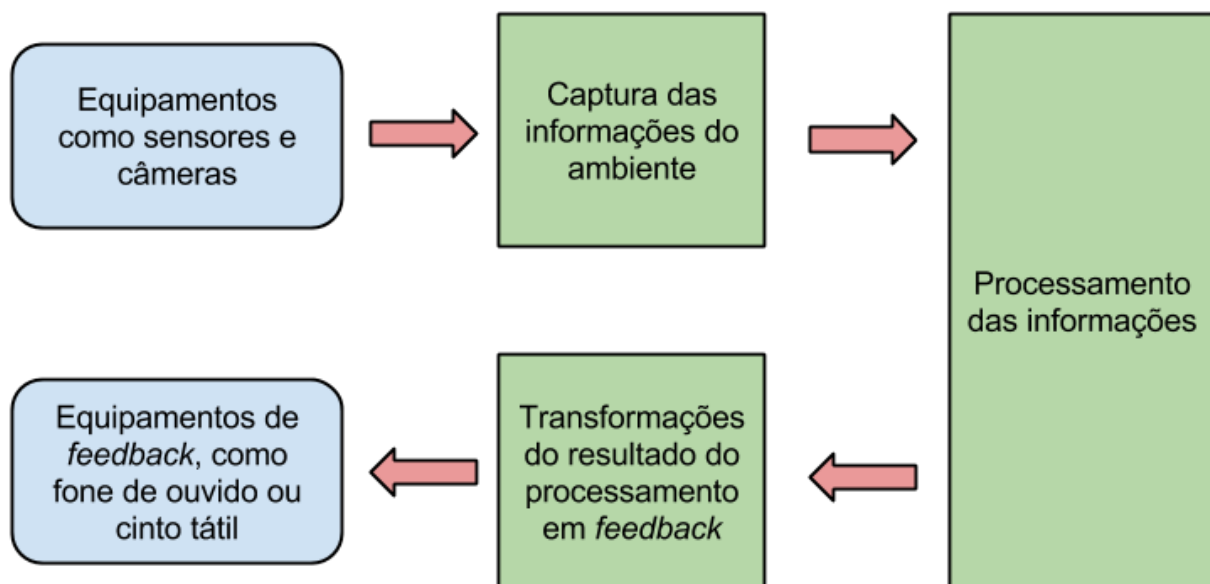


Figura 2.1 – Conjunto de etapas padrão identificadas nos modelos apresentados

Para captura de informações, os protótipos, em geral, utilizam equipamentos *off-the-shelf*, isto é, equipamentos prontos e de preferência de fácil acesso, como sensores, fones de ouvido, câmeras, *smartphones*, entre outros. O processamento das informações capturadas se dá através do uso de várias técnicas e algoritmos, tais como, visão computacional e aprendizagem de máquina. Quanto ao *feedback* para o usuário, dependendo da complexidade da informação e do objetivo do modelo, normalmente, é feito por meio sonoro e ou tátil.

Com *feedbacks* sonoros a informação pode ser muito mais específica e de melhor entendimento, em contrapartida diminui ou oculta do usuário o som ambiente. Como o som ambiente, em geral, é uma informação útil para um usuário com deficiência visual, para minimizar este problema, o trabalho de Shoval et al. [37] utiliza fones de osso-condução.

Estes fones ficam apoiados sobre ossos da cabeça a fim de transmitir o som, e mantém os ouvidos do usuário livres para os sons ambientes.

Os *feedbacks* táteis não limitam o som do ambiente para o usuário, entretanto, conseguem transmitir menos informação em relação aos *feedbacks* sonoros e necessitam um tempo maior de aprendizagem pelo usuário. Com o intuito de melhorar esta interface, vários trabalhos tem apresentado novas formas de *feedback* tátil.

3. MODELO DE APOIO À NAVEGAÇÃO EM CALÇADAS

Nesta capítulo é apresentado o modelo proposto, incluindo a metodologia utilizada para o seu desenvolvimento.

3.1 Metodologia

Inicialmente, foi realizado um estudo de trabalhos relacionados ao auxílio na mobilidade de pessoas com deficiência visual. Como descrito no capítulo 2, diversas propostas de modelos de auxílio à locomoção de pessoas com deficiência visual foram apresentadas ao longo dos últimos anos. Uma parte das pesquisas destina-se à detecção e desvio de obstáculos [37, 4, 25, 6, 1, 42, 32, 22] e, em geral, nestes modelos um dos objetivos é demonstrar uma nova forma de *feedback* para o usuário. Há também pesquisas para detectar faixas de pedestres [23, 44, 39], seguir pisos táteis [26, 24], detectar escadas [44], navegar em ambientes internos [14, 18, 7, 37], entre outras [36, 8, 5, 26, 21, 9, 35, 43].

Embora muitos avanços tenham sido publicados nesta área de auxílio à navegação autônoma para pessoas com deficiência visual, ainda existem problemas de pesquisas para serem resolvidos que envolvem, por exemplo, tratamento de imagem, identificação de padrões e processamento de sinal digital [29].

Segundo o Instituto Ethos [12], são muitas as necessidades de pessoas com deficiência visual, que incluem: identificar outras pessoas; identificar a existência de degraus, pisos escorregadios, buracos e outros obstáculos durante o trajeto; conhecer a direção que deve tomar e as distâncias para o destino; entre outras.

Portanto, este estudo forneceu a base para: formular a pergunta de pesquisa, definir o objetivo geral deste trabalho, e identificar alguns tópicos de pesquisas que poderiam apoiar à navegação em calçadas para pessoas com deficiência visual. Estes tópicos são apresentados na lista seguir:

- a) alerta sobre a existência de piso tátil de alerta e direcional;
- b) alerta sobre a existência de obstáculos terrestres;
- c) alerta sobre a existência de obstáculos aéreos;
- d) auxílio para a localização de faixas de pedestres;
- e) auxílio na locomoção sobre faixas de pedestres;
- f) alerta sobre sinalerias em faixas de pedestres;
- g) informações sobre o entorno;
- h) auxílio na rota e identificação de distâncias.

Certamente esta lista não é completa, mas serviu como base para definir as partes que compõe este trabalho, bem como para identificar possíveis trabalhos futuros. Sendo assim, os quatro primeiros itens da lista (letras *a*, *b*, *c* e *d*) foram escolhidos para fazerem parte deste trabalho. Esta escolha foi feita a partir de estudos quanto às técnicas que poderiam ser utilizadas para solução de cada problema, pesquisas prévias realizadas durante o período de estudos e a dimensão do trabalho para ser realizado no período predeterminado.

O modelo geral proposto está ilustrado na figura 3.1. Após, iniciou-se uma pesquisa individual para cada uma das partes que o compõem, analisando trabalhos com propósitos semelhantes e/ou técnicas que pudessem ser utilizadas. A finalidade era encontrar abordagens que pudessem ser melhoradas e ainda que pudessem ser posteriormente integradas.

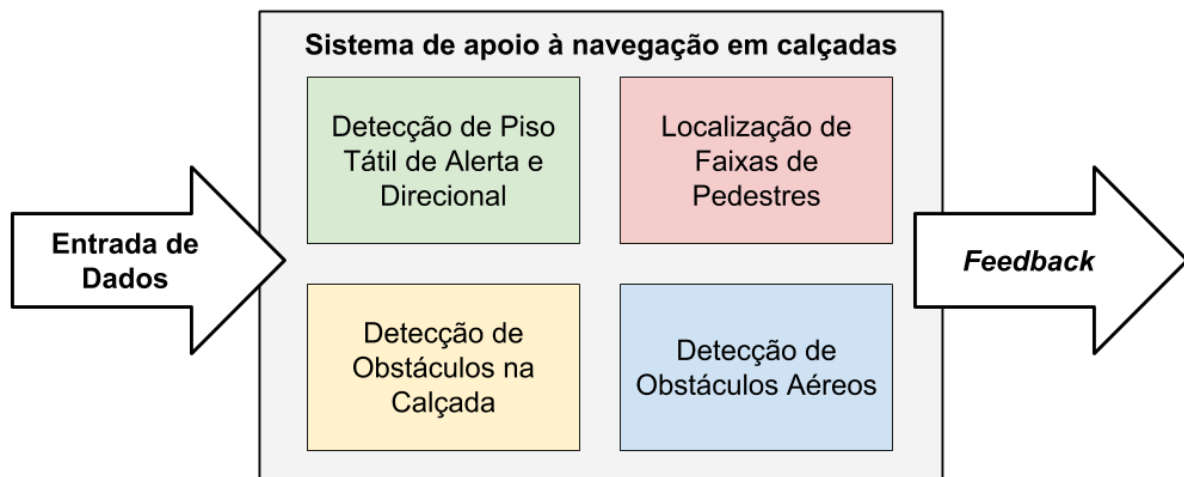


Figura 3.1 – Modelo geral proposto

Conforme a pesquisa individual para cada parte era finalizada, um protótipo com o algoritmo desenvolvido era implementado e testado para posterior integração no modelo proposto. Ou seja, a integração da detecção de pisos táteis e obstáculos terrestres ocorreu depois dos seus protótipos terem sido desenvolvidos individualmente. O modelo proposto conta com um *feedback* sonoro que permite fornecer as informações desejadas para o deficiente visual, podendo auxiliar, assim, no seu deslocamento em calçadas.

3.2 Descrição do Modelo

Detecção de pisos táteis é a primeira parte que compõe o modelo. Piso tátil corresponde às faixas em alto relevo fixadas no chão para fornecer auxílio a pessoas com deficiência visual. Existem dois tipos de piso tátil: o direcional (Figura 3.2(a)), que informa o caminho mais seguro; e o de alerta (Figura 3.2(b)), que informa locais que necessitam

de maior atenção. No Brasil, estes pisos táteis são encontrados com diversos padrões de cores, como ilustra a figura 3.3.



Figura 3.2 – Exemplo de pisos táteis: (a) Piso Direcional; (b) Piso de Alerta.



Figura 3.3 – Modelos de cores de pisos táteis.

Seguir uma faixa de piso tátil direcional é, na teoria, uma forma de garantir que o usuário mantenha-se sobre a calçada em um caminho seguro. Avisar ao usuário sobre a existência de pisos táteis de alerta também fornece mais segurança na sua navegação, uma vez que ele indica a existência de uma entrada de garagem ou uma esquina, por exemplo.

Deteção de obstáculos terrestres é a segunda parte do modelo, que é totalmente integrada com a deteção de pisos táteis. A necessidade de seu desenvolvimento é porque no Brasil ainda são poucas as calçadas que possuem pisos táteis. Assim, as pessoas com deficiência visual precisam caminhar sobre calçadas que muitas vezes possuem vários obstáculos. Em geral, estes obstáculos são detectados por pessoas com deficiência visual através bengala longa, todavia, a possibilidade de antecipar o perigo pode ser um benefício para estas pessoas. No modelo proposto, considera-se obstáculo em calçada qualquer elemento que possa impedir ou prejudicar a passagem. Por exemplo, buracos, canteiros ou degraus, ou também quando há uma parede, interseção de uma rua, entre outros, como ilustrado na figura 3.4.

O processo de deteção de piso tátil e obstáculos na calçada está representado, em alto nível, no algoritmo 3.1. Após capturar a imagem da calçada que está a frente do

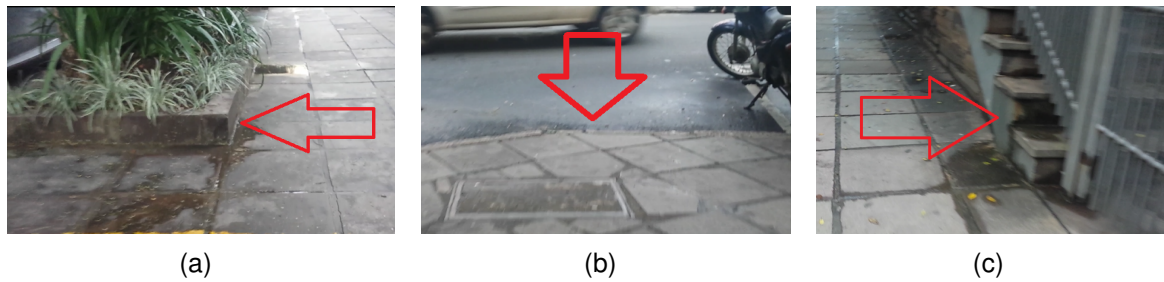


Figura 3.4 – Exemplos de obstáculos em calçadas: (a) Canteiro; (b) Esquina; (c) Parede.

usuário (I_f), pesquisa-se pela presença de piso tátil direcional (P_d). Se encontrado, obtém-se a sua direção e o usuário é guiado para permanecer sobre o piso. Portanto, se ele estiver saindo de cima do piso tátil, é passado um aviso de que lado está este piso. Caso exista um piso de alerta (P_a), é gerado um *feedback* sonoro ao usuário. Na falta de pisos táteis, procura-se por obstáculos na calçada (O_c). São identificados padrões de texturas nas calçadas, e se houver um local com padrão divergente do padrão de onde o usuário se encontra, é emitido um *feedback* de alerta ao usuário.

Algoritmo 3.1 – Processo de detecção de piso tátil e obstáculos na calçada.

```

1: while (Video Ativo) do
2:    $I_f$  é o próximo frame com imagem da calçada
3:   Pesquisa por Piso Direcional na imagem
4:    $P_d$  é resultado da pesquisa por piso direcional
5:   if  $P_d = \text{SIM}$  then
6:     informe ao usuário a direção do piso
7:   else
8:     Pesquisa por Piso de Alerta na imagem
9:      $P_a$  é resultado da pesquisa por piso de alerta
10:    if  $P_a = \text{SIM}$  then
11:      informe ao usuário sobre o alerta
12:    else
13:      Pesquisa por obstáculos na calçada na imagem
14:       $O_c$  é resultado da pesquisa por obstáculos na calçada
15:      if  $O_c = \text{SIM}$  then
16:        informe ao usuário sobre o obstáculo
17:      end if
18:    end if
19:  end if
20: end while

```

Uma descrição da implementação e dos resultados preliminares sobre a pesquisa de detecção de pisos táteis e detecção de obstáculos em calçadas, é feita nos capítulos 4 e 5, respectivamente.

A detecção de obstáculos aéreos é a terceira parte que compõe o modelo. A partir de uma conversa informal com o Sr. Daniel Bernadini, presidente da ADVBG (Associação dos Deficientes Visuais de Bento Gonçalves), identificou-se que outra dificuldade de pessoas com deficiência visual são os obstáculos aéreos. Como demonstrado no capítulo 2, a detecção de obstáculos aéreos não é uma área de pesquisa nova, mas é muito importante para auxiliar o deslocamento de pessoas com deficiência visual. Por este motivo, definiu-se que o modelo proposto deve incorporar um mecanismo para detecção de tais obstáculos.

Para isto, optou-se pelo uso de um sensor ultrassônico, que pode fornecer a distância de obstáculos em relação ao usuário. Quando o sensor detecta um obstáculo a uma determinada distância do usuário, é possível emitir alertas para ele.

O processo de detecção de obstáculos aéreos está representado, em alto nível, no algoritmo 3.2. Considera-se que o sensor está sempre ativo, e ao obter a distância (S_d) para algum objeto que esteja na sua frente é verificado se esta distância é menor que a distância de aviso de parada (P_d). Se a resposta for positiva, é emitido um aviso para o usuário parar imediatamente. Senão, é verificado se a distância (S_d) é menor que a distância de aviso de alerta (A_d) para emitir um aviso de alerta de obstáculos ao usuário.

Algoritmo 3.2 – Processo de detecção de obstáculos aéreos.

```

1: while (Sensor Ativo) do
2:    $S_d$  é a distância retornada pelo(s) sensor(es)
3:    $P_d$  é a distância de aviso de parada
4:    $A_d$  é a distância de aviso de alerta
5:   if  $S_d \leq P_d$  then
6:     Emite alerta de parada obrigatória
7:   else
8:     if  $S_d \leq A_d$  then
9:       Emite alerta de obstáculo
10:    end if
11:  end if
12: end while

```

Implementação do protótipo de obstáculos aéreos está descrito no capítulo 4 na seção 4.4.

A localização de faixas de pedestres é a quarta e última parte que compõe o modelo proposto. Fornecer a localização e a distância das faixas de pedestres, assim como, a distância das esquinas, é uma informação importante para o usuário planejar sua navegação, além de auxiliá-lo a atravessar ruas com mais segurança.

A pesquisa de faixas de pedestres e esquinas é realizada, utilizando técnicas de visão computacional, em imagens de satélite e de mapas rodoviários da localidade do usuário. Com o advento dos *smartphones* equipados com GPS e acesso a internet, tornou-se fácil a obtenção das coordenadas geográficas e destas imagens, exemplificadas na figura 3.5.



Figura 3.5 – (a) Imagem de satélite; (b) imagem rodoviária.

Algoritmo 3.3 – Processo para obter as distâncias de esquinas e faixas de pedestres.

- 1: U_c é a posição geográfica do usuário
- 2: Obtém-se as imagens de satélite e do mapa rodoviário em U_c
- 3: S_m é a imagem de satélite obtida
- 4: R_m é a imagem do mapa rodoviário obtida
- 5: Encontra esquinas na imagem rodoviária R_m
- 6: E_d é o resultado da distância das esquinas em relação ao usuário
- 7: **if** $\text{Count}(E_d) > 0$ **then**
- 8: Informa ao usuário a distância das esquinas E_d
- 9: **else**
- 10: Informa ao usuário que não há esquinas próximas
- 11: **end if**
- 12: Pesquisa faixas de pedestres na imagem de satélite S_m
- 13: F_d é a relação de faixas de pedestres próximas ao usuário
- 14: **if** $\text{Count}(F_d) > 0$ **then**
- 15: Informa ao usuário a distância para as faixas de pedestres
- 16: **else**
- 17: Informa ao usuário que não há faixas de pedestres
- 18: **end if**

O processo para obter estas informações está representado, em alto nível, no algoritmo 3.3. O processo é iniciado manualmente pelo usuário, e então o sistema obtém a sua posição geográfica (U_c). A partir desta posição, são buscadas a imagem de satélite (S_m) e a imagem do mapa rodoviário (R_m) do local correspondente a posição U_c . Pesquisa-se na imagem do mapa rodoviário as esquinas da quadra onde o usuário se encontra (E_d) e para informá-lo a distância das mesmas. Então, pesquisa-se na imagem de satélite se há faixas de pedestres próximas ligadas a quadra na qual o usuário se encontra (F_d). Se encontrar, são informadas a direção e distância das faixas de pedestres.

Uma descrição da implementação e dos resultados preliminares sobre a pesquisa de obtenção de distâncias de esquinas e faixas de pedestres também é feita nos capítulos 4 e 5, respectivamente.

4. IMPLEMENTAÇÃO DO MODELO

Nesta capítulo é apresentada a implementação do modelo proposto. A primeira seção descreve sua entrada de dados e as demais seções os algoritmos desenvolvidos para cada parte que compõe o modelo.

4.1 Entrada de Dados

Considerando a entrada de dados do modelo, a detecção de pisos táteis e de obstáculos terrestres funciona tanto com vídeos pré-gravados, quanto por entrada de vídeo direto, como com uma câmera de *smartphone* ou uma *webcam*, com resolução igual ou superior a 320×240 pixels em orientação de paisagem.

Quando a orientação da imagem estiver em retrato corta-se a parte superior da imagem transformando a orientação para paisagem, se a resolução permitir. A fim de que o modelo proposto possa executar em tempo real, optou-se por este ganho de velocidade em detrimento da acurácia. Se a imagem possuir resolução maior que 320×240 pixels, diminuiu-se a sua qualidade, mantendo a proporção, a fim de aproximá-la da resolução desejada. A modificação da qualidade da imagem pode provocar a perda de dados, mas a resolução utilizada reduz o tempo de processamento que os algoritmos necessitam para percorrer a imagem. Esta resolução foi escolhida por ser a menor resolução que levou à geração de bons resultados, ou seja, melhor acurácia sem perda de desempenho. Por isso, o algoritmo não foi avaliado em imagens com resolução maiores.

Como ilustrado na figura 4.1, considera-se que todas as imagens são obtidas a uma altura média de 1 metro e ângulo em torno de 45° , ou seja, um pouco acima da cintura de uma pessoa de estatura mediana. Com este posicionamento obtém-se imagens com uma visão bem próxima aos pés do usuário e com mais de 3 metros de profundidade.

A imagem, já com a resolução desejada, é convertida para o espaço de cor YCbCr¹ [33]. Este espaço de cor pode ser usado para separar o componente de luminância (canal Y), ou brilho, de uma imagem a partir de suas cores, permitindo que o algoritmo processe imagens menos suscetível a incidência de sombras e problemas de iluminação. Além disso, ajuda a reduzir a variância global de cores entre o mesmo espectro de luz. Por exemplo, azul claro e azul marinho, embora ambas as cores estejam no espectro azul, sua variação é grande o suficiente para afetar os padrões de algoritmos de reconhecimento de imagem. Após o ajuste de resolução e a conversão para o espaço de cor YCbCr a imagem está pronta para o processamento dos algoritmos.

¹O espaço de cor YCbCr é um modelo matemáticos usado para descrever a cor de pixels de imagens

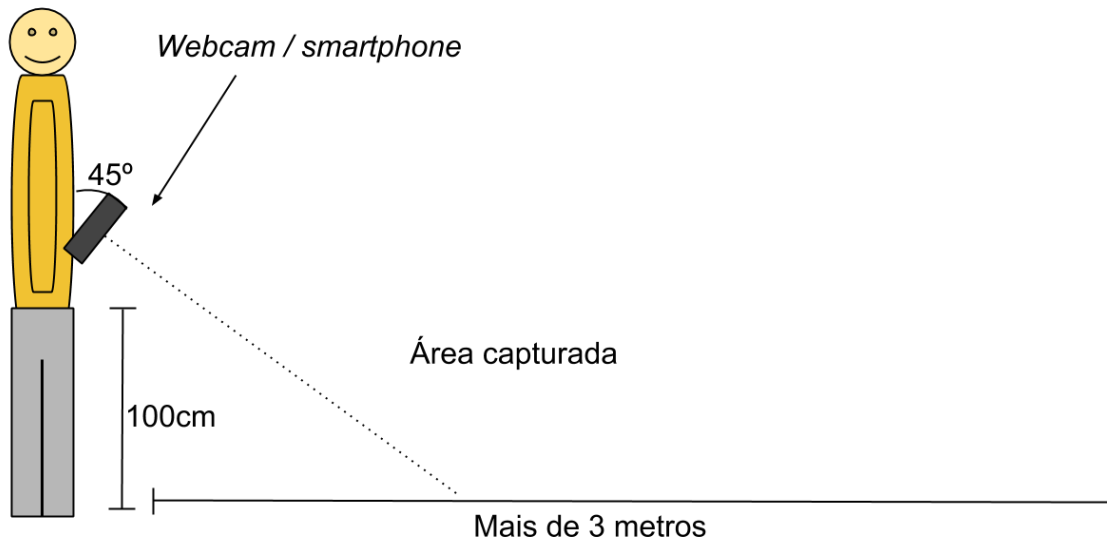


Figura 4.1 – Ilustração da aquisição da imagem.

Para a detecção de obstáculos aéreos um sensor ultrassônico foi usado para entrada de dados. A informação de distância é passada via *bluetooth* para um dispositivo que a interpreta, como ilustra a figura 4.13.

No caso da identificação e localização de faixas de pedestres e esquinas, a entrada de dados corresponde a uma coordenada de GPS obtida por um *smartphone*. A partir desta coordenada, são obtidas as imagens de satélite e rodoviária da região correspondente através da API do Google Maps. Como foi usada uma versão gratuita desta API, cada imagem obtida (de satélite e rodoviária) é uma matriz com resolução de 640×640 pixels, capturada por um zoom de $20\times$ (ou seja, a altura da imagem em relação ao solo), correspondente a uma área de $83m^2$, aproximadamente.

4.2 Pisos Táteis

O algoritmo desenvolvido para detecção de pisos táteis é composto por várias etapas, como pode ser observado na figura 4.2 e foi desenvolvido com base no trabalho de Jie et al.[24]. Como descrito na 4.1, o primeiro e o segundo passo consistem, respectivamente, no ajuste do tamanho imagem e na sua transformação para o espaço de cor YCbCr. Em seguida, inicia-se o processamento para detectar a superfície de piso tátil, que numa primeira etapa encontra e aplica limiares para extrair a área de interesse da imagem, eliminando zonas que não podem ser piso tátil.

Depois disto, ocorre a detecção de bordas e a identificação das linhas retas relacionadas a estas bordas. A área entre duas linhas é analisada para verificar se é de um padrão aceitável em relação a superfície de piso tátil. A imagem resultante é, em seguida,

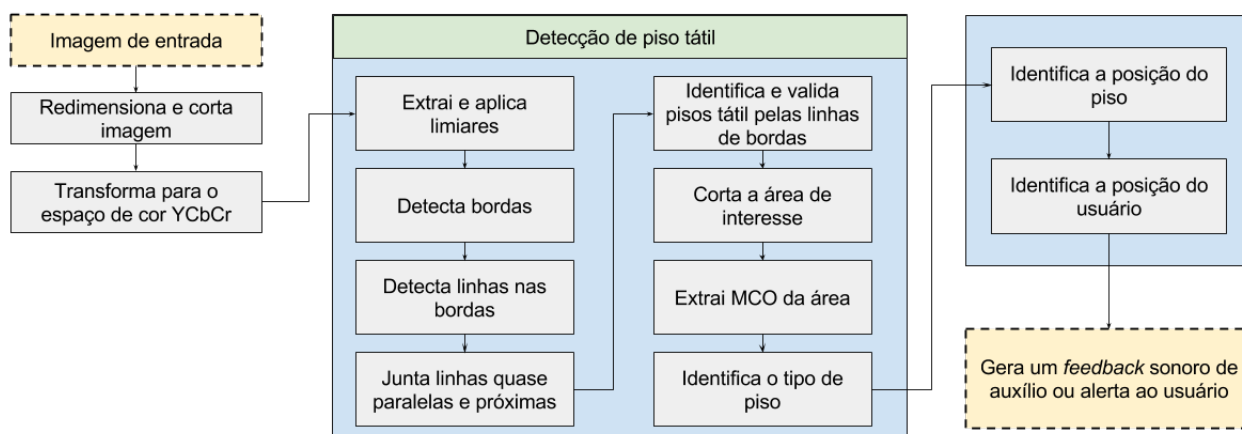


Figura 4.2 – Abordagem para a detecção de piso tátil.

segmentada em vários pequenos blocos. Oito matrizes de co-ocorrência (GLCM)[11] são processadas para cada bloco (quatro direções e duas distâncias). Os valores de entropia, contraste, homogeneidade e uniformidade são calculados para cada GLCM.

Os resultados, então, são aplicados a uma Árvore de Decisão[13] que identifica se é piso tátil e de qual tipo, direcional ou de alerta. Finalmente, um *feedback* de som é emitido para alertar o usuário se há um piso tátil, seu tipo, e sua posição em relação ao centro do piso. A seguir uma descrição detalhada destas etapas apresentadas na figura 4.2.

Após a preparação da imagem, um histograma para cada canal (Y, Cr e Cb) é gerado e normalizado com os valores variando entre 0 e 255. O histograma das imagens (figura 4.3) apresenta picos relativos às cores das superfícies de pisos táteis. Estes picos combinados com algoritmo de limiar são utilizados para determinar na imagem a possível área de piso tátil. Uma vez que pisos táteis possuem padrão de cores diferentes, foram definidas duas configurações para representar suas áreas no histograma, chamadas de *Tipo A* e *Tipo B*. *Tipo A* funciona melhor com cores mais vivas como amarelo ou azul, e *Tipo B* funciona melhor com cores opacas como cinza. Este processo gera um total de 6 valores limiares para os 3 canais da imagem (Min/Max Y, Min/Max Cr e Min/Max Cb).

Os limiares utilizados são dinâmicos, uma vez que são obtidos através de fórmulas definidas empiricamente. Os valores definidos para o processo de obtenção de limiares foram encontrados após várias heurísticas. As fórmulas podem ser observadas na tabela 4.1.

Um resultado da aplicação dos limiares pode ser visto na figura 4.4(b). Depois, então, são aplicados operadores morfológicos de erosão e dilatação nestas imagens a fim de eliminar possíveis falsos positivos. Um resultado da aplicação destes operadores é mostrado na figura 4.4(c). Além disso, se a quantidade de pixels detectada como área de interesse é menor do que 10% ou maior do que 40% da quantidade total de pixels da imagem, a imagem é marcada como sem piso tátil.

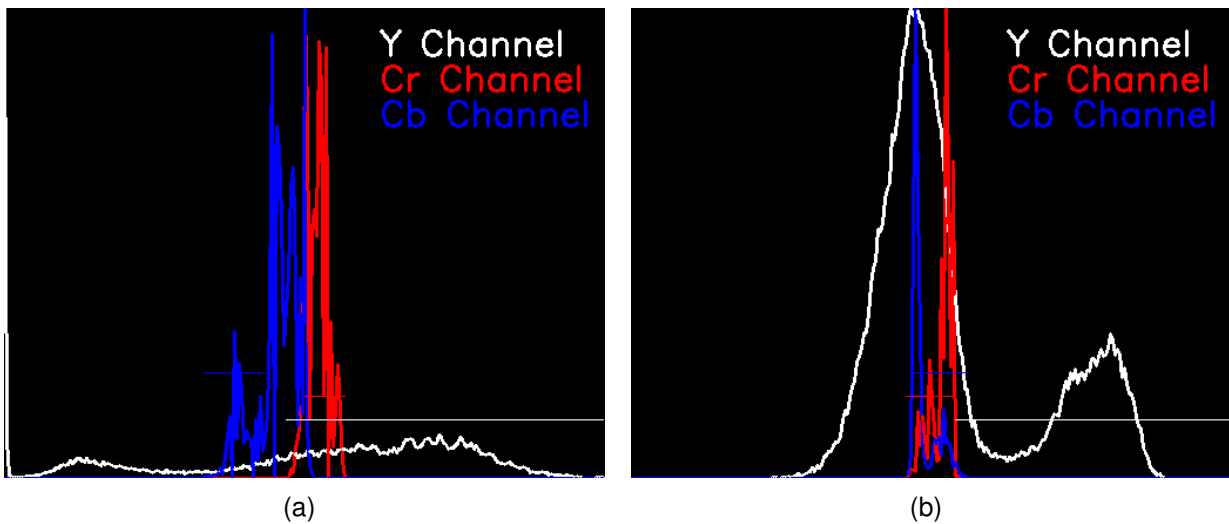


Figura 4.3 – Histogramas de pisos táteis: (a) tipo A; (b) tipo B.

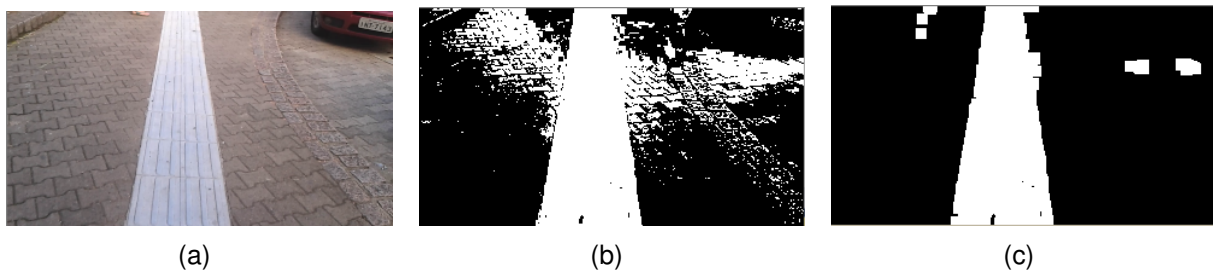


Figura 4.4 – Exemplo da aplicação dos limiares e operadores morfológicos: (a) Imagem Original; (b) Após limiares; (c) Após operações morfológicas.

Ao aplicar os limiares de *Tipo A* ou *Tipo B*, uma possível área de piso tátil é extraída da imagem. Um exemplo da área detectada por este processo pode ser observado na figura 4.5. O uso do *Tipo A* ou *Tipo B* no algoritmo é determinado pela falta de detecção de “linhas paralelas” na imagem. Neste trabalho, foram consideradas como linhas “paralelas”, duas ou mais linhas que apontem para a mesma direção, mas que não são necessariamente equidistante, sendo capaz de se cruzarem em algum ponto da imagem. O algoritmo começa usando o limiar de *Tipo A* em um *frame*, se forem detectadas “linhas paralelas” mantém neste tipo, caso contrário, muda para o limiar de *Tipo B*. No entanto, após a primeira detecção, se um *frame* não detectar linhas paralelas, mantém o mesmo limiar para até três *frames*, só então, se ainda não voltou a detectar, muda o limiar. Esse recurso de manter o limiar por três *frames* tem como objetivo manter a detecção de piso tátil mesmo em falsos negativos, mas sem permitir que o usuário ande sobre possíveis falsos positivos por muito tempo.

As “linhas paralelas” detectadas para piso tátil são, geralmente, linhas retas longas. Para detectar estas “linhas paralelas” é utilizado o detector de bordas Canny [33]. Como ele produz um melhor resultado se a imagem não possui ruídos, um filtro de desfocagem

Tabela 4.1 – Fórmula para formação de limiares.

Tipo	Limiar	Fórmula
A	Min Cb	Sem limiar.
	Max Cb	É a primeira posição com valor acima de 40 antes do segundo pico do histograma Cb.
	Min Cr	A partir do primeiro pico no histograma Cr, é a primeira posição na qual o valor é menor que o valor máximo do primeiro pico do histograma Cr ou o valor é menor que 135.
	Max Cr	Sem limiar.
	Min Y	Sem limiar.
	Max Y	Sem limiar.
B	Min Cb	A partir do primeiro pico no histograma Cb, é a primeira posição na qual o valor é menor que 40.
	Max Cb	Sem limiar.
	Min Cr	A partir do maior pico no histograma Cr, é a primeira posição na qual o valor for inferior a 135.
	Max Cr	Sem limiar.
	Min Y	A partir do primeiro pico no histograma Y, é a primeira posição na qual o valor é menor que 50.
	Max Y	Sem limiar.

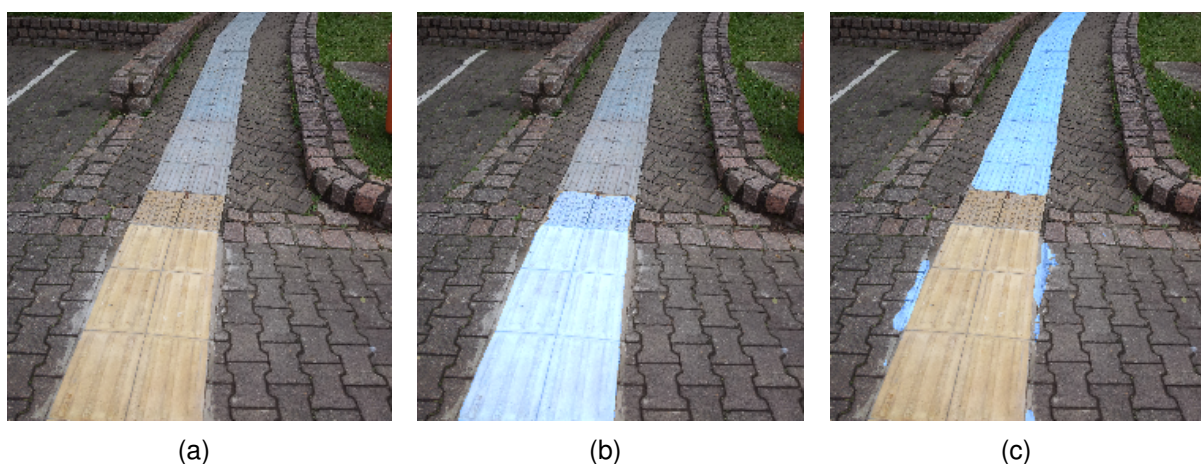


Figura 4.5 – Exemplo de área detectada: (a) Original; (b) Tipo A; (c) Tipo B.

(*median blur*) é aplicado à imagem antes de executar o algoritmo de detecção de bordas para reduzir o ruído. Em seguida, é aplicada na imagem resultante a transformada de Hough [33] para detecção das linhas. Os resultados destes passos podem ser vistos na figura 4.6.

No algoritmo de transformada de Hough é possível definir níveis de limiar de detecção, e quanto maior o nível, menos linhas são detectadas. Como cada linha detectada precisa ser analisada, quanto mais linhas, menor é o desempenho. Todavia, quando se utiliza níveis altos, pisos táteis irregulares, sujos ou altamente segmentados, param de ser detectados. Considerando obstáculos no caminho, curvas abruptas ou mesmo um pequeno

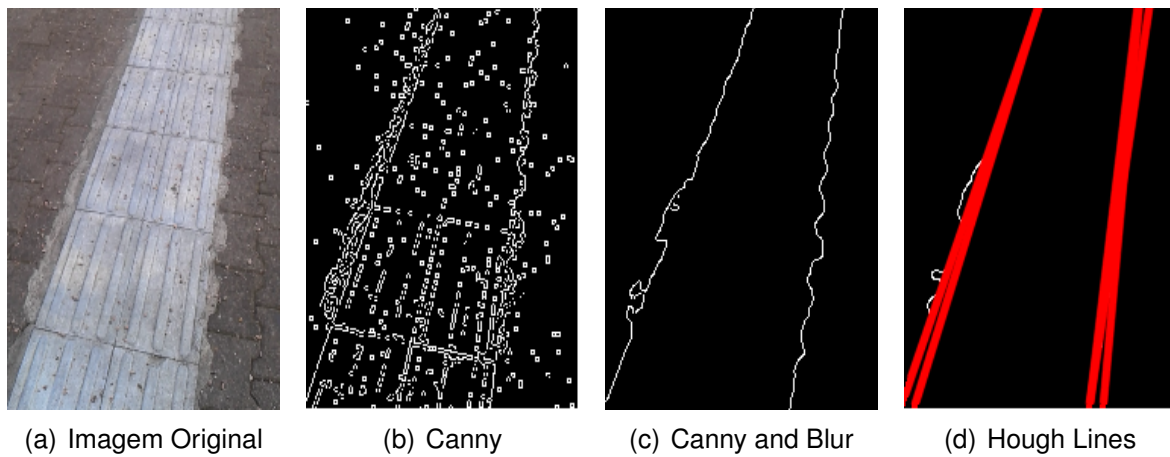


Figura 4.6 – Processo de detecção de linhas

desvio do usuário em relação ao piso tátil, e depois de um estudo dos níveis de limiares, o nível 25 provou ser capaz de detectar pisos táteis curtos e longos sem muitas falhas. Este valor foi, então, escolhido como o nível do limiar aplicado na transformada de Hough.

Muitas linhas detectadas se sobrepõem e representam a mesma linha na imagem original, mas com ângulos ligeiramente diferentes. Então, para ter um melhor desempenho e resultado, foi implementada uma função que mescla as linhas detectadas que estão muito próximas e que possuem ângulos semelhantes. Mesmo assim, ainda pode existir mais do que duas linhas nos resultados. Neste caso, são escolhidas as duas linhas “paralelas” cuja a distância entre elas é compatível com o espaço necessário para um piso tátil. Como o posicionamento da câmera é fixo, é possível definir limites para tamanho de piso tátil. Verifica-se o ângulo θ (é o ângulo em relação ao eixo x, resultado da transformada de Hough) em cada linha e as linhas cujo θ diferem entre 0, 10 e 0, 45 são consideradas “linhas paralelas”. Estes valores foram escolhidos depois de testar vários valores diferentes. Também é considerada a distância das linhas em relação ao usuário, isto é, na base da imagem (próximo do usuário) a distância entre as linhas precisa ser maior do que na parte superior da imagem.

O par de “linhas paralelas” com os menores valores de θ e que são compatíveis com piso tátil, são, então, selecionados como a fronteira do piso tátil. Embora esse método ainda produza alguns falsos positivos, é o melhor método encontrado em comparação com outras soluções como, por exemplo, simplesmente selecionando as duas “linhas paralelas” maiores ou a aplicação de um preenchimento para verificar se o polígono resultante corresponderia a um piso tátil. Depois de detectadas as fronteiras, calcula-se também o centro e direção deste piso.

Porém, a seleção das “linhas paralelas” não garante que as linhas sejam fronteiras de um piso tátil. Depois de testes em diversos vídeos e imagens tornou-se clara a necessidade da textura da imagem resultante ser analisada. Assim, a imagem resultante é

separada em blocos de 25×25 pixels. Oito matrizes de co-ocorrência (GLCM) são processadas para cada bloco (quatro direções por duas distâncias). Para cada GLCM gerada, são extraídos os valores de entropia, contraste, homogeneidade e uniformidade (ou energia).

Embora o detector de bordas Canny foi usado antes de aplicar a transformada de Hough para encontrar as linhas “paralelas”, ele não se mostrou adequado para a detecção de pequenas variações. Portanto, após a análise de alguns detectores, optou-se por utilizar o detector de bordas Laplace[33] antes de aplicar a GLCM. Uma comparação dos métodos pode ser observada nas imagens da figura 4.7.

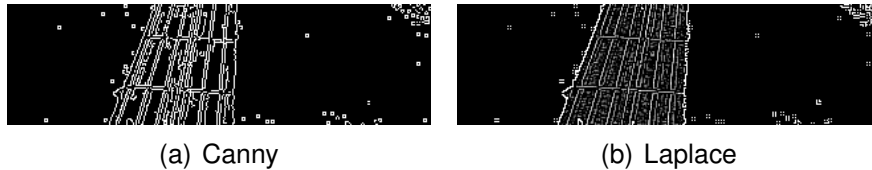


Figura 4.7 – Comparação entre diferentes detectores de bordas.

Para melhorar o desempenho, blocos que têm mais do que 75% de pixels pretos são descartados automaticamente (blocos em branco). A divisão de um bloco pode ser observada na Figura 4.8.

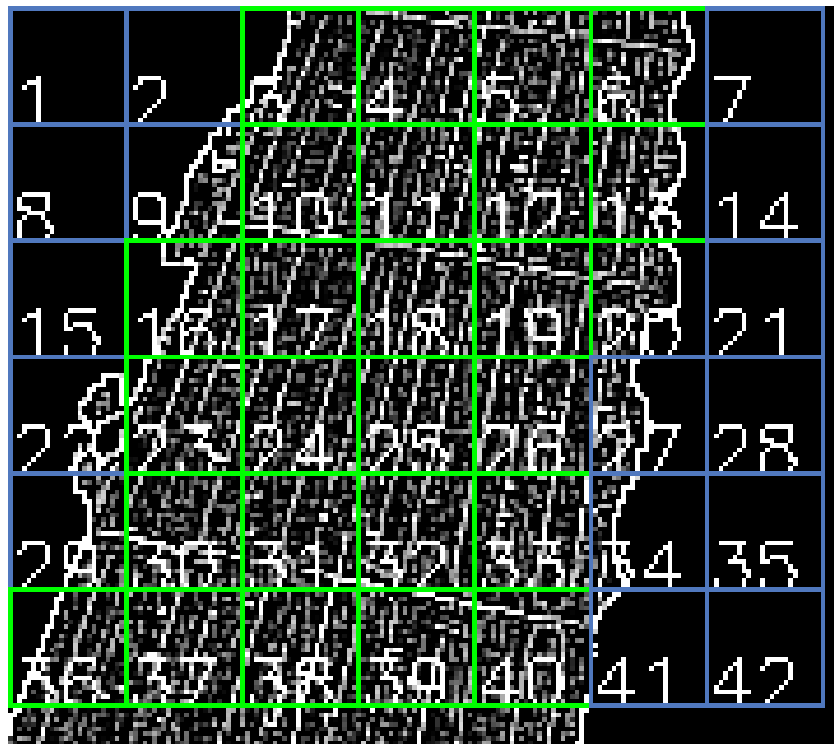


Figura 4.8 – Bloco Segmentado.

Usando o minerador de dados RapidMiner², foi extraída uma Árvore de Decisão que permite prever o tipo de bloco (Alerta, direcional ou ruído). O resultado, que consiste

²<https://rapidminer.com/products/studio/>

na árvore de decisão, foi escrito como um algoritmo do sistema através de comandos condicionais (*ifs*) encadeados, sem a necessidade de utilizar bibliotecas externas ou qualquer outro algoritmo.

Para treinar esta Árvore de Decisão foram utilizadas 624 entradas de diferentes tipos de blocos, e para determinar o tipo da imagem foram utilizadas as seguintes regras:

- **Alerta:** Se dois blocos vizinhos são do tipo de alerta, a imagem é considerada como contendo um piso de alerta;
- **Direcional:** Se a primeira regra não se aplica e um quadrado 2x2 blocos são do tipo direcional, a imagem é considerada como contendo um piso direcional;
- **Ruído:** Se nenhuma das regras anteriores se aplica, a imagem é considerada sem piso tátil.

Essas regras ajudam a reduzir a quantidade ocorrências de falsos positivos e, ao mesmo tempo, preservam a qualidade da detecção de pisos táteis. Para melhor exemplificar, um exemplo de piso de alerta com ruídos pode ser observado na figura 4.9(a) e os blocos analisados podem ser vistos na figura 4.9(b). Já a figura 4.9(c) contém um exemplo de piso direcional com ruídos cuja os blocos analisados podem ser vistos na figura 4.9(d).

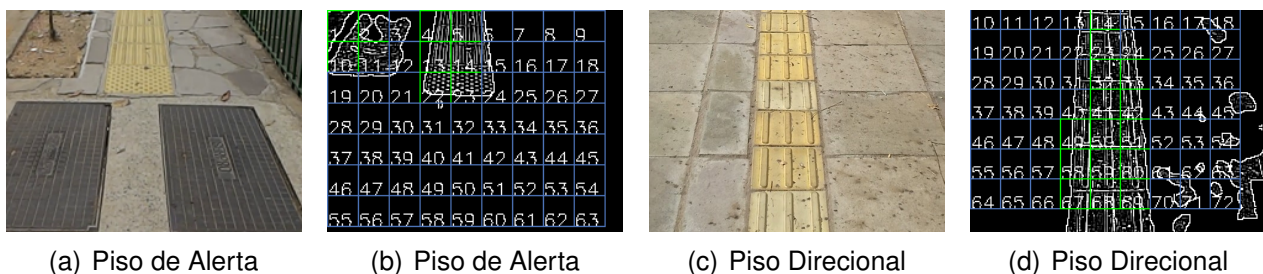


Figura 4.9 – Exemplo de pisos táteis com ruídos.

Após toda a análise da imagem o sistema retorna um dos três possíveis resultados: piso tátil direcional, piso tátil de alerta ou ruído (sem piso tátil). Se um piso for encontrado, fornece o correto *feedback* ao usuário através da emissão de sons, ou beeps. São utilizados três sons diferentes, aqui chamados de **Beep de Alerta**, **Beep de Direção**, e **Beep de Piso**.

Se o resultado for piso tátil direcional, calcula-se a posição do usuário em relação ao piso e o *feedback* segue a seguinte regra: Se o usuário está sobre o piso, emite o som **Beep Piso**; se o piso está a direita do usuário, emite o som **Beep Direção** no ouvido direito; se o piso está a esquerda do usuário, emite o som **Beep Direção** no ouvido esquerdo. Se o resultado for piso tátil de alerta, o *feedback* será a emissão do som **Beep Alerta**. Na falta de um piso tátil o processo é direcionado ao algoritmo de obstáculos em calçadas, descrito na próxima seção.

Esta implementação foi desenvolvida para detecção de pisos táteis encontrados em calçadas durante o dia. Mas, o modelo proposto também poderia ser utilizado no período noturno, quando há menos iluminação, e para pisos de outros formatos, desde que fossem feitas alterações no algoritmo. Por exemplo, poderia ser utilizada uma câmera com infravermelho para o caso de haver pouca iluminação e poderia ser feito um novo treinamento da árvore de decisão para detecção de outros tipos de piso táteis.

4.3 Obstáculos Terrestres em Calçadas

A detecção de obstáculos terrestres em calçadas é uma sequência da detecção de pisos táteis (seção 4.2). Esta parte possui menos etapas que a anterior, como pode ser observado na figura 4.10.

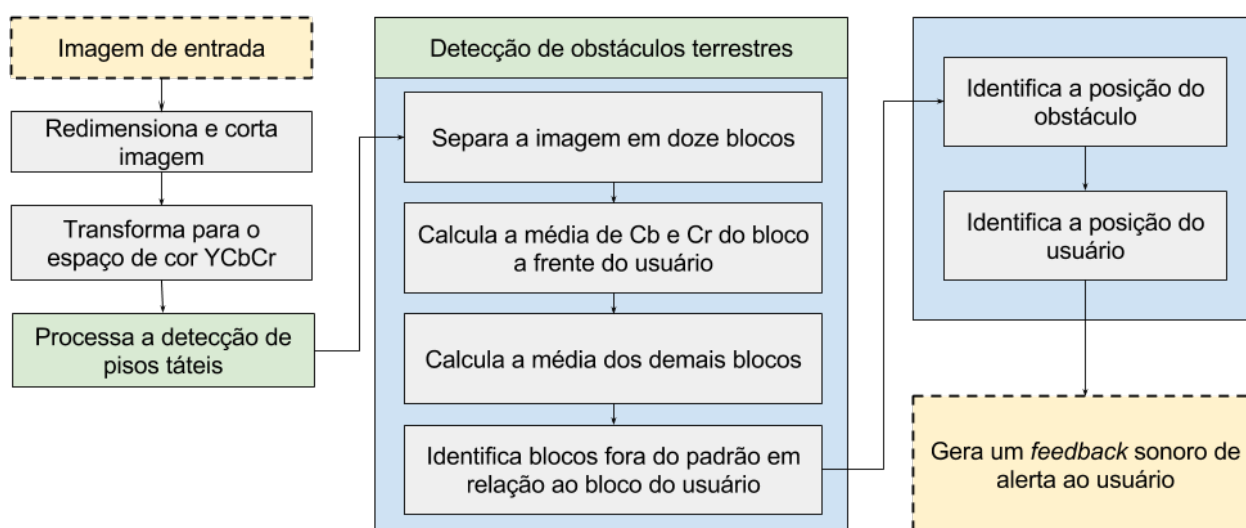


Figura 4.10 – Abordagem para a detecção de obstáculos terrestres.

Nesta parte, optou-se por trabalhar com a detecção de mudanças de padrão de calçadas, mais especificamente, inconsistência ou mudança no padrão de cores. Esta ideia se baseia na hipótese de que uma inconsistência no padrão de cores da calçada pode significar um obstáculo. Este processo não detecta todos obstáculos, mas pode ser processado rapidamente e utiliza a mesma imagem já pré-processada para a detecção de pisos táteis.

Analisando visualmente imagens de calçadas, identificou-se que a maioria dos obstáculos possuem um padrão de cores diferente do restante da calçada. Na figura 4.11 pode-se ver um canteiro e a parte de um carro que são exemplos de obstáculos.

O primeiro passo, portanto, foi o ajuste da imagem descrito na 4.1. Depois, a imagem é dividida em 12 blocos, como mostra a figura 4.11(a). Como a imagem é em perspectiva, mesmo os blocos tendo o mesmo tamanho em pixels, as áreas reais de cada



Figura 4.11 – Exemplo de imagem com obstáculos: (a) Imagem dividida em blocos; (b) Características de cores de cada bloco.

bloco são de tamanhos diferentes umas das outras. Optou-se por esta divisão porque os blocos mais próximos do usuário correspondem a medidas próximas a 90cm , e em média o passo humano tem em torno de 85cm , ou seja, assim, pode-se prever obstáculos um passo a frente do usuário.

O bloco 8 na figura 4.11(a) é o mais próximo do usuário, e por isso é considerado como bloco referência da posição do usuário. Os demais blocos, quando possuem características diferentes do bloco referência, são considerados obstáculos.

Na figura 4.11(a), pode-se ver dois obstáculos, um canteiro, localizado nos blocos 1 e 2, e parte de um carro, localizado no bloco 9. A figura 4.11(b) corresponde a mesma imagem da figura 4.11(a), porém, cada bloco está colorido pela média de sua cor. Ao analisar visualmente a figura 4.11(b), considerando como referência o bloco 8, é possível ver uma diferença maior de cor nos blocos 1, 2 e 9, que correspondem aos obstáculos.

Para cada bloco é calculado o valor médio dos canais **Cb** e **Cr**. Estes valores são exemplificados na figura 4.11(b). Os valores **Dif Cb** e **Dif Cr** representam, respectivamente, a diferença dos canais **Cb** e **Cr** do bloco em relação ao bloco 8. Como explicado anteriormente, o canal **Y** não é usado porque este canal representa a luminosidade, a exclusão deste canal permite eliminar diferenças causadas por luminosidade e sombras.

Ao fazer uma análise numérica, observa-se que os valores absolutos de **Dif Cb** e **Dif Cr**, em sua maioria, são inferiores a $0,01$, exceto nos blocos 1, 2 e 9, nos quais pelo menos um dos valores é maior ou igual a $0,01$, o que indica que estes são os blocos que correspondem a áreas que possuem obstáculos. O mesmo experimento foi realizado e analisado em um total de 50 imagens de calçadas diferentes, quando observou-se que dias de muita luminosidade solar influenciam negativamente no resultado.

Um fator importante para a velocidade de processamento é selecionar apenas as áreas de interesse para o usuário, isto é, áreas próximas a ele. As imagens capturadas, como demonstrado anteriormente, possuem mais de 3 metros de profundidade. Sendo assim, apenas os blocos que estão pintados com a cor cinza na figura 4.12 são processados.

Quando um destes blocos está fora do padrão do bloco de referência é emitido um alerta para o usuário. Na figura 4.12 o bloco que está fora do padrão aparece marcado como “CUIDADO” e o bloco de referência está marcado como “USUÁRIO”.



Figura 4.12 – Blocos da imagem que são processados.

O *feedback* para o usuário é fornecido através da emissão de sons (beep), assim como no *feedback* da detecção de piso tátil. Neste caso, é utilizado apenas o som chamado de **Beep Alerta**. Sempre que o algoritmo processar um bloco que for considerado como um possível obstáculo é emitido o som **Beep Alerta**.

4.4 Obstáculos Aéreos

Na detecção de obstáculos aéreos, a maior complexidade está na montagem e programação do equipamento. O trabalho de Valentim [42] foi usado como base, mas apesar dele utilizar três sensores, optou-se pelo uso de apenas 1 sensor como teste de conceito, com possibilidade de inclusão de mais sensores posteriormente.

Como ilustra a figura 4.13, o sensor é acoplado ao usuário na altura da cabeça. As informações de distância de objetos a frente do sensor são enviadas, via *bluetooth*, à um dispositivo que interpreta a informação e, se necessário, converte a informação em um *feedback* sonoro ao usuário.

O protótipo foi projetado considerando que o sensor deve ser acoplado a um chapéu ou boné, como ilustra a figura 4.14, de forma que o usuário fique com as mãos livres. Para isto, foram utilizados os seguintes dispositivos:

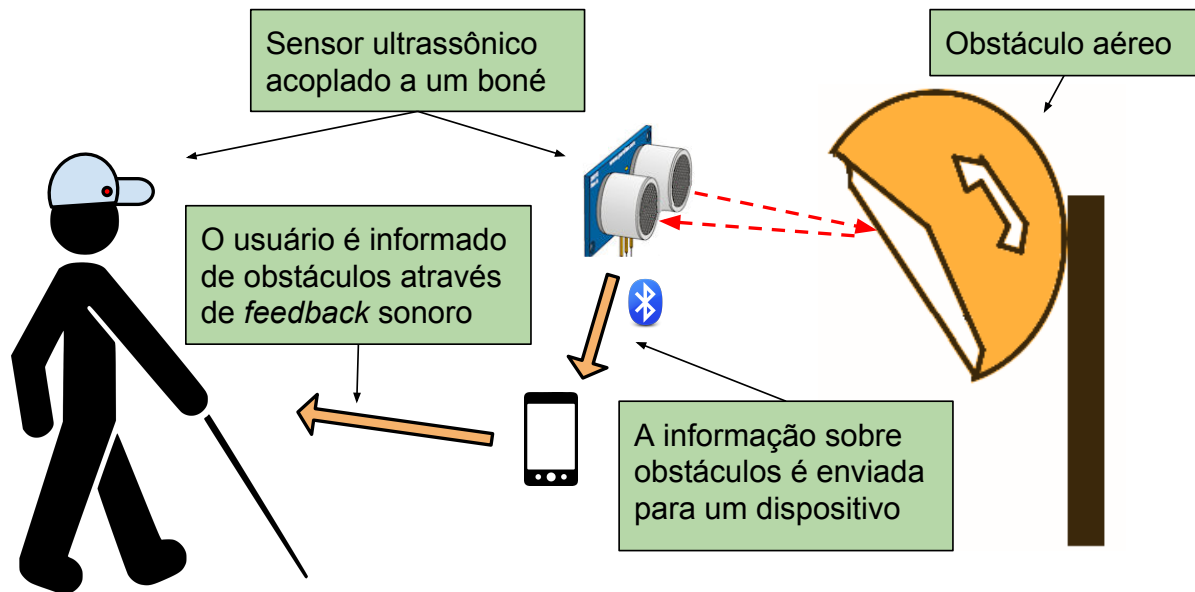


Figura 4.13 – Detecção de obstáculos aéreos do modelo proposto.

1. Um sensor ultrassônico modelo HC-SR04 [2] utilizado para detectar obstáculos e retornar suas distâncias. É um sensor de baixo custo capaz de medir distâncias entre 2cm a 4m com boa precisão de obstáculos em um ângulo de 15°.
2. Um módulo *bluetooth* JY-MCU modelo HC-05 utilizado para transmitir as informações do sensor ultrassônico para um *smartphone*. Este módulo transmite e recebe informações via *bluetooth*³ a uma distância de até 10m e velocidade de até 2,1Mbps.
3. Uma placa Arduino UNO R3 responsável por integrar o sensor ultrassônico com o módulo *bluetooth*. Arduino é uma plataforma de prototipagem eletrônica de hardware livre, o que facilita a integração de componentes eletrônicos.
4. Uma bateria utilizada como alimentador de energia e fios para integrar todos eletrônicos.
5. Um *smartphone* que recebe as informações, processa, e retorna o *feedback* sonoro para o usuário.

O sensor ultrassônico está ligado, por fio, a uma placa Arduino com módulo *bluetooth* e bateria portátil. Esta placa foi programada para capturar, a cada 500ms, as informações do sensor para enviar, via *bluetooth*, a outro equipamento, neste caso, um *smartphone*. A programação da placa Arduino está representada, em alto nível, no algoritmo 4.1.

Para receber as informações e dar o *feedback* para o usuário foi utilizado um *smartphone* com sistema operacional Android. A programação do *smartphone* está repre-

³*Bluetooth* é o nome dado à tecnologia de comunicação sem fio de que permite transmissão de dados e arquivos de maneira rápida e segura através de dispositivos digitais.

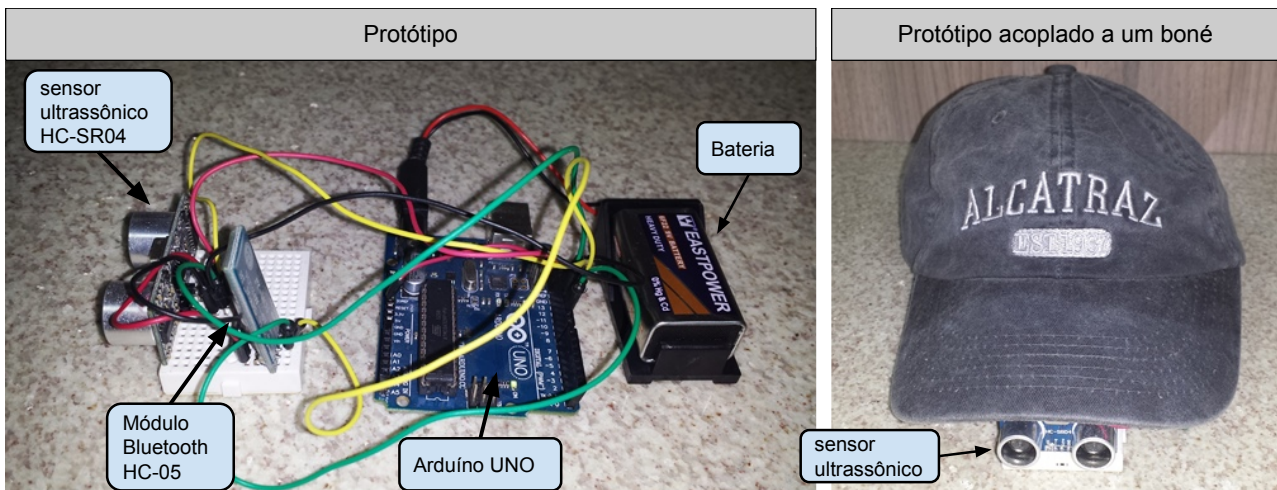


Figura 4.14 – Equipamento para detecção de obstáculos aéreos.

Algoritmo 4.1 – Integração entre o sensor ultrassônico e o módulo *bluetooth*.

- 1: B_t é a conexão ativa do *bluetooth*
- 2: Inicia o módulo *bluetooth*
- 3: Inicia o sensor ultrassônico
- 4: **loop**
- 5: Aguarda conexão por *bluetooth* em B_t
- 6: **while** (B_t) **do**
- 7: Recebe informações de distâncias do sensor
- 8: Transmite informações recebidas para B_t
- 9: Aguarda 500ms
- 10: **end while**
- 11: **end loop**

sentada, em alto nível, no algoritmo em 4.2. Foi escrito um *script* para QPython⁴. Este *script* se conecta, via *bluetooth*, com a placa Arduino, depois fica aguardando pelo valores de distâncias. Quando recebe uma distância verifica se é menor que 1 metro para emitir um *feedback* ao usuário dizendo “PARE”. Caso contrário, se a distância for maior que 1 metro e menor de 2 metros, emite um *feedback* ao usuário dizendo “CUIDADO”, e se for maior que 2 metros não faz nada.

Como o sensor utilizado é de baixo custo, detecta obstáculos em uma angulação de 15 graus e está localizado sobre a cabeça do usuário, são detectados apenas obstáculos de altura próxima a cabeça do usuário. Para uma detecção de uma área maior, é necessário utilizar mais sensores ultrassônicos. Além disso, o sensor possui bons resultados em ambiente controlado, retornando a distância com precisão de milímetros, porém, em ambiente externo apresenta uma grande quantidade de falhas e valores falsos.

⁴QPython é um motor de *script* que executa *scripts* Python em dispositivos Android.

Algoritmo 4.2 – Recepção das informações e *feedback* para o usuário.

```

1: Cria uma conexão com o módulo bluetooth
2:  $B_t$  é a conexão ativa do bluetooth
3:  $O_d$  é distância recebida por bluetooth
4: while ( $B_t$ ) do
5:   Recebe distância em  $O_d$ 
6:   if ( $O_d < 1$  metro) then
7:     emite uma mensagem de voz: PARE
8:   else
9:     if ( $O_d < 2$  metros) then
10:      emite uma mensagem de voz: CUIDADO
11:    end if
12:  end if
13: end while

```

4.5 Faixas de Pedestres e Esquinas

O modelo para auxiliar o usuário a localizar faixas de pedestres e obter as distâncias de esquinas está representado na figura 4.15 e cada uma das etapas que aparecem nesta figura estão descritas a seguir:

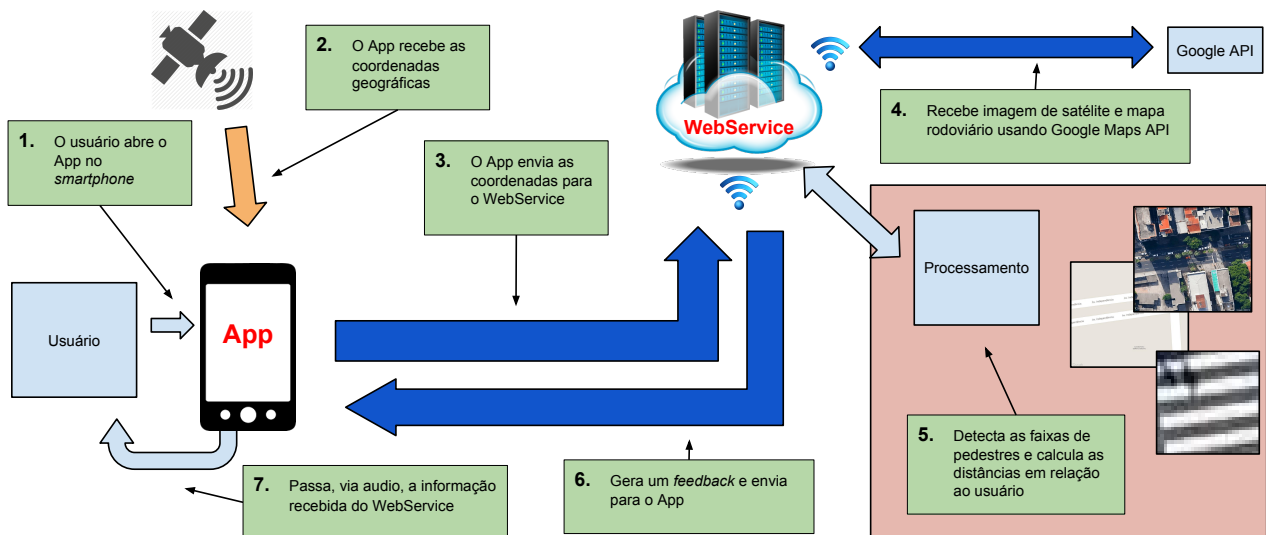


Figura 4.15 – Localização de faixas de pedestres e esquinas do modelo proposto.

1. O usuário abre o App em um *smartphone* que deve ter tecnologia GPS e acesso a internet;
2. O App obtém as coordenadas geográficas do usuário;
3. O App envia as coordenadas geográficas para o *Webservice*;
4. O *Webservice* acessa um serviço de mapas e obtém mapas correspondentes às coordenadas geográficas;

5. Utilizando algoritmos de visão computacional e uma SVM⁵ (*Support Vector Machines*) [13], extrai das imagens a localização de faixas de pedestres e a distância para as esquinas quando existirem;
6. O *WebService* devolve ao App as informações obtidas, o endereço, a distância das esquinas em metros e a distância de faixa de pedestres quando existirem;
7. O *feedback* para o usuário se dá por voz.

Conforme mencionado na seção 4.1, o primeiro passo está relacionado com a aquisição das coordenadas GPS. Estas coordenadas são usadas para inicializar o processo de obtenção de imagens e identificar a relação entre o usuário e as faixas de pedestres detectadas. As coordenadas geográficas, capturadas pelo *smartphone* do usuário, são enviadas para um serviço Web. O serviço Web é responsável por receber as coordenadas geográficas do local do usuário (ilustrado na figura 4.16(a) por um ponto vermelho) e inicializar a obtenção das imagens. A API do Google Maps é usada para obter a imagem de satélite (Figura 4.16(b)) e do seu respectivo mapa rodoviário (Figura 4.16(c)). Estas imagens são usadas no processo de detecção de faixas de pedestres e esquinas e o resultado, ou seja, a distância das esquinas e faixas de pedestre, é fornecido ao usuário por áudio. A seguir, os principais passos deste processo são detalhados.



Figura 4.16 – A aquisição das imagens: (a) Coordenadas GPS enviadas pelo usuário, ilustrada pelo ponto vermelho; (b) Imagem de satélite extraída do Google Maps; (c) imagem rodoviária extraída do mesmo conjunto de dados.

O módulo de processamento de imagem é responsável por extrair as imagens do Google Maps, bem como por orientar a detecção das faixas de pedestres. O primeiro passo do módulo de processamento de imagem (após a aquisição) refere-se a segmentação da imagem. A segmentação da imagem consiste em selecionar as regiões de interesse (ruas e calçadas). O objetivo desta abordagem é o de melhorar o desempenho computacional, bem como eliminar a possibilidade de encontrar falsos positivos em áreas sem interesse (por exemplo, topo de edifícios, coberturas, etc.).

⁵SVM-Support vector machines: modelo de aprendizagem de máquina

Inicialmente, a imagem de satélite e a imagem rodoviária (designado como **R**), são convertidas em tons de cinza. A região de interesse **B** é gerada através da limiarização de cada pixel (x, y) de **R** por λ (onde $\lambda = 245$, definida experimentalmente), conforme especificado na Equação 4.1. A figura 4.17(a) ilustra o resultado da abordagem de limiar.

$$B(x, y) = \begin{cases} 0, & \text{if } R(x, y) \leq \lambda \\ 1, & \text{caso contrário} \end{cases} \quad (4.1)$$

Como se pode ver na figura 4.17(a), existem várias estruturas indesejáveis na imagem binária (texto, setas, etc). Tentando eliminar tais estruturas indesejáveis, uma operação morfológica que consiste em uma combinação de erosões e dilatações, é aplicada a esta imagem binária. Mais precisamente, primeiro aplicam-se duas erosões na imagem **B**, com um elemento estruturante de tamanho 3×3 . A imagem resultante é, então, dilatada 20 vezes, usando o mesmo elemento estruturante. O objetivo deste procedimento é remover pequenos artefatos encontrados na imagem binária e expandir a área de interesse para lidar com desalinhamento do mapa rodoviário (no que diz respeito à imagem de satélite). A ordem, número e tipo de operações morfológicas, foram definidas experimentalmente.

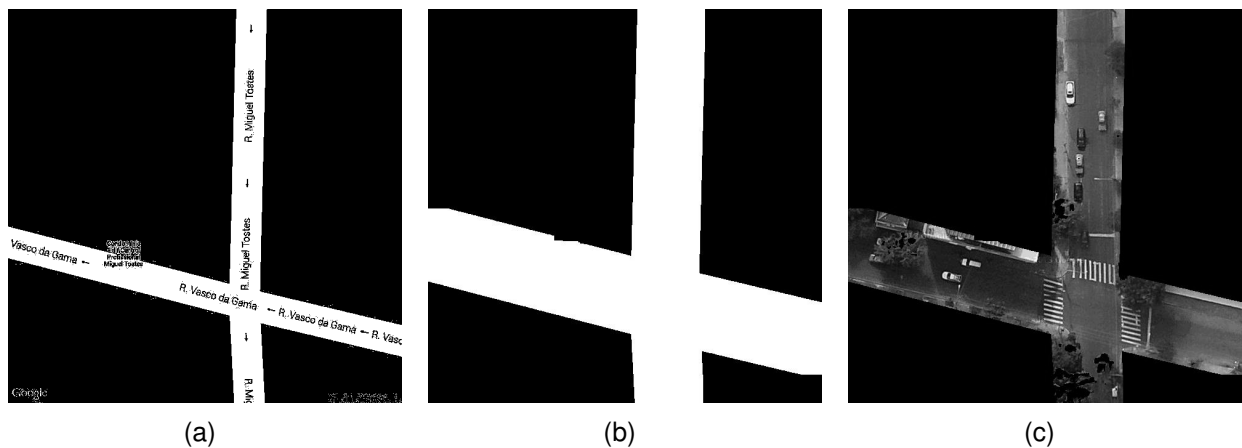


Figura 4.17 – Segmentação da imagem: (a) Segmentação inicial na imagem rodoviária; (b) Saída após operações morfológicas - Imagem binária; (c) Junção da imagem de satélite com as regiões de interesse.

A saída das operações morfológicas, ilustrada na figura 4.17(b), é então combinada com a respectiva imagem de satélite em tons de cinza (Figura 4.17(c)). Este procedimento é usado nas próximas etapas para detecção de faixas de pedestres, como descrito a seguir.

Neste trabalho, foi treinado um classificador SVM para detectar faixas de pedestres (padrão de “zebra”) a partir de imagens de satélite. O conjunto de dados para treino é composto por 370 blocos de imagens de padrão de zebra (amostras positivas) e 530 blocos de imagens de padrão não zebra (amostras negativas), como exemplificado na figura 4.18. Os blocos foram extraídos manualmente a partir das imagens de satélite obtidas através da

API do Google Maps (considerando as imagens de entrada descritas na seção 4.1), sendo que cada bloco possui o tamanho 30×30 pixels.

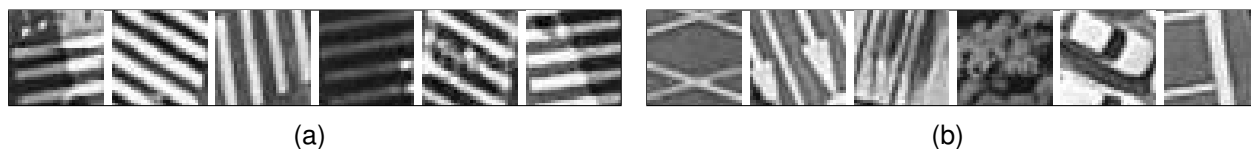


Figura 4.18 – Exemplo da extração manual de amostras (a) positivas e (b) negativas.

Como se pode ver na figura 4.18, os blocos variam em condições de iluminação (por exemplo, capturados em diferentes momentos do dia, com e sem sombras), orientação, ângulo e formas (em relação ao padrão de zebra). Além disso, alguns blocos de imagens positivas são parcialmente obstruídos por pessoas, carros e/ou árvores, bem como alguns blocos de imagens negativas contêm objetos com linhas paralelas, tais como setas direcionais. A ideia de incluir esses recursos no conjunto de dados de aprendizagem é para lidar com diferentes situações/cenários que normalmente encontram-se em aplicações reais.

Para o treinamento do classificador SVM, as imagens do banco de dados foram divididas aleatoriamente de forma estratificada em um conjunto de dados de treinamento (contendo 600 imagens) e um conjunto de dados para teste independente (com 300 imagens). O conjunto de treinamento foi utilizado em uma validação cruzada de 10 vezes (*10-fold*) para avaliar o desempenho do SVM para estimar os parâmetros. Uma vez que os parâmetros ideais foram encontrados, um novo modelo foi treinado usando o conjunto completo de dados de treinamento e aplicando ao conjunto de teste independente.

Neste trabalho utilizou-se o operador LBP⁶ (Local binary patterns) [33] como método de extração de características de texturas para treinar o SVM. LBP é um operador de textura simples, mas muito eficiente que rotula os pixels de uma imagem, limiariza os pixels vizinhos e considera o resultado como um número binário [11]. Ele pode ser utilizado para reconhecer uma grande variedade de tipos de textura, em que os métodos estatísticos e estruturais convencionalmente têm sido usadas separadamente. Também tentou-se usar para extração de características o operador GLCM e uma combinação de ambos. A avaliação dos diferentes métodos de extração de características de textura foi demonstrada no capítulo 5 seção 5.2.

Depois de treinar o classificador SVM, a etapa seguinte do modelo proposto é a detecção de faixas de pedestres. Considere uma imagem de entrada de satélite, delimitada pela região de interesse, tal como ilustrada na figura 4.17(c). Esta imagem de entrada é dividida em células pequenas, com tamanho igual a 15×15 pixels, como mostrado na figura 4.19(a) (ilustrado por uma região recortada, destacada na figura 4.19(d), para fins de visualização).

⁶LBP é um tipo de descritor visual utilizado para a classificação em visão computacional

Na fase de classificação, quatro células vizinhas são agrupadas para formar um bloco com tamanho igual a 30×30 (relacionado com o mesmo tamanho dos blocos usados para treinar nosso classificador SVM), ilustrado na figura 4.19(a) por um quadrado vermelho. Cada bloco é composto pela célula de referência (a célula superior esquerda, destacada na figura 4.19(b) por um quadrado verde) e suas células vizinhas (isto é, a direita, abaixo e abaixo da célula da direita). Tal procedimento é realizado para cada célula que está na região de interesse. O objetivo deste procedimento é criar uma região de sobreposição entre os blocos adjacentes para lidar com desalinhamentos na fase de classificação, bem como para executar uma verificação de dois passos, como descrito a seguir.

Se um bloco de 30×30 é considerado faixa de pedestres, uma segunda verificação é realizada. Nesta segunda verificação, 8 amostras vizinhas são criadas em torno do bloco de referência (as células de referência destes 8 blocos estão ilustradas na figura 4.19(c) por quadrados azuis). Se todos esses oito blocos vizinhos também são considerados uma faixa de pedestres, o bloco de referência é então definido como faixa de pedestres, caso contrário o bloco de referência é descartado.

Utilizando a imagem de satélite (Figura 4.19(d) como referência, a figura 4.19(e) ilustra as regiões das faixas de pedestres informadas manualmente por um usuário, enquanto a figura 4.19(f) ilustra o resultado de detecção após o processamento pelo modelo proposto, os blocos detectados como faixas de pedestres são ilustrados por blocos brancos.

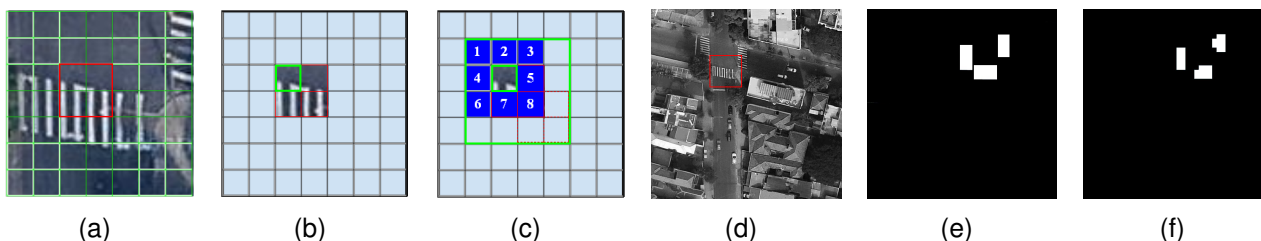


Figura 4.19 – Detecção de faixas de pedestres: (a) Região recortada de uma imagem em análise com um bloco de referência 30×30 marcado por um quadrado vermelho; (b) Bloco de referência e sua célula de referência 15×15 destacada por um quadrado verde; (c) Células de referência utilizadas na segunda verificação; (d) Imagem de entrada satélite (com a região recortada ilustrada na imagem (a) em destaque); (e) *Ground truth* informado manualmente por um usuário; (f) Estimativa final, dado pelo modelo proposto.

Dadas as faixas de pedestres detectadas, o próximo estágio do modelo proposto refere-se a fazer uma relação espacial entre o usuário e as faixas de pedestres (ou seja, a localização de faixas de pedestres em relação ao usuário). Em primeiro lugar, se o usuário está no meio de uma quadra, o modelo assume que ele está de frente para a rua, desta forma, fica simples de indicar a localização de faixas de pedestres, isto é, pode-se informar que a faixa está à direita, à esquerda ou à frente do usuário, como ilustrado na figura 4.20. Se o usuário está em uma esquina, assume-se que ele está virado de frente para a mesma, a localização da faixa é fornecida da mesma forma, à direita ou à esquerda do usuário.

Seguindo a linha da rua em ambas as direções (esquerda e direita), como mostrado na figura 4.20, analisando a imagem binária (Figura 4.17(c)), pode-se definir a quadra onde o usuário está localizado e encontrar as esquinas (se existir), assim como, seleccionar as faixas de pedestres detectadas que estão conectadas à quadra do usuário. As esquinas também podem ser usadas para dar ao usuário um *feedback* adicional, quando uma faixa de pedestre não é detectada informa-se as distâncias para as esquinas.

O objetivo é informar ao usuário a faixa de pedestres mais próxima (ou as duas mais próximas quando ambas estão a distâncias iguais do usuário). Como já explicado, duas situações diferentes são considerados: **i)** o usuário está posicionado na esquina; **ii)** o usuário está voltado para a rua (por exemplo, posicionado no meio da quadra). O importante é definir um ponto de referência utilizado para informar ao usuário a localização de faixas de pedestres (por exemplo, há uma faixa de pedestres a sua esquerda).

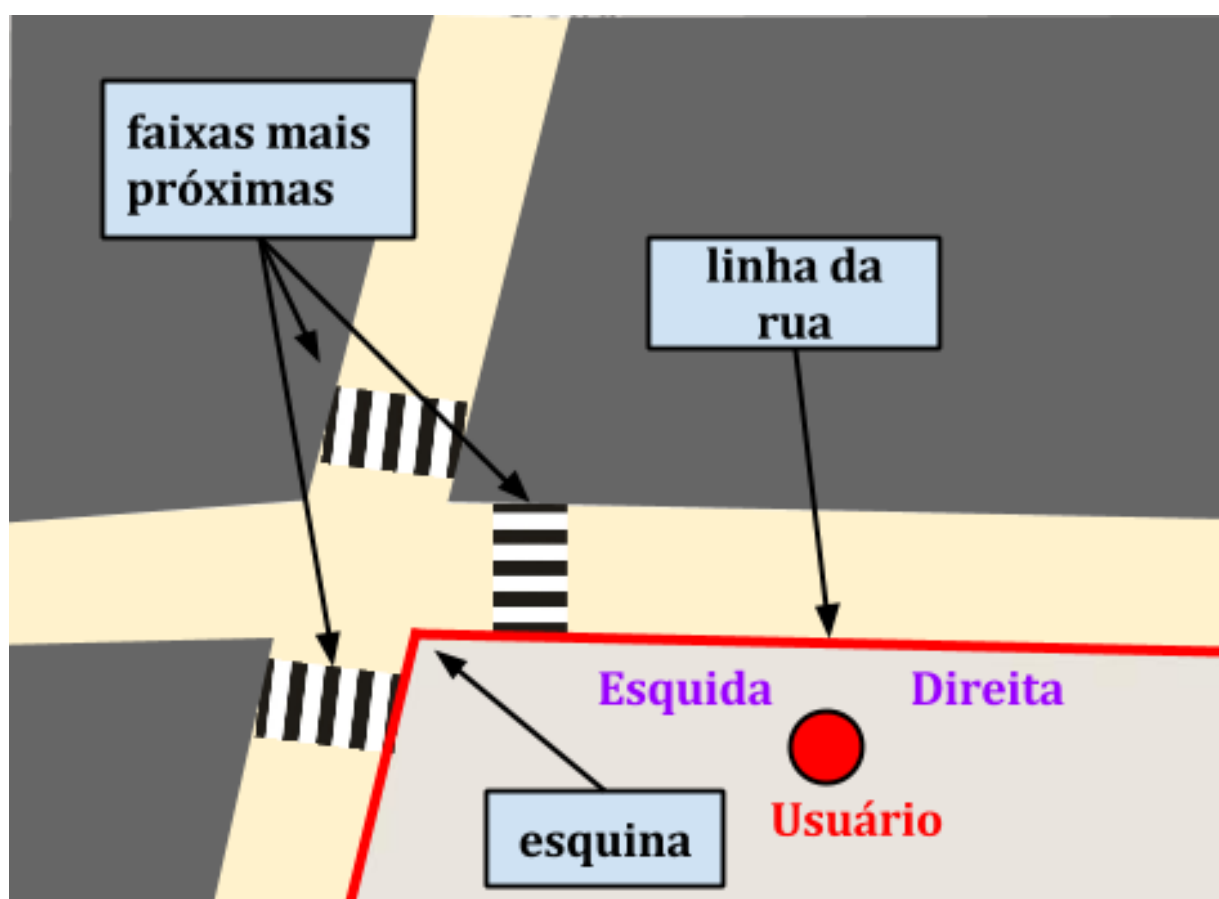


Figura 4.20 – Ilustração da linha da rua, esquinas e faixas de pedestres ligadas a quadra onde o usuário está.

Assim, se forem detectadas duas ou mais faixas, em primeiro lugar calcula-se a distância do usuário para cada uma delas e para as esquinas. A distância entre o usuário e a faixa de pedestres mais próxima (ou esquina) é calculada usando uma relação muito simples, que leva em consideração a resolução da imagem e sua real dimensão extraída da API do Google Maps. Como se sabe que cada imagem tem 640 pixels de largura e

altura, relacionados a 82,69 metros quadrados, pode-se supor que cada pixel representa 0,129203125 metros. No caso de duas ou mais faixas terem aproximadamente a mesma distância em relação ao usuário (foi definida como distâncias, em relação ao usuário, que diferem menos de 3 metros), ele é informado sobre as duas mais próximas. Esse procedimento é descrito no algoritmo 4.3. Esta informação é usada para calcular o *feedback* final, o qual é enviado para o usuário, como explicado a seguir.

Algoritmo 4.3 – Localização de Faixas de Pedestres.

- 1: N_c é o número de faixas de pedestres encontrado na quadra
- 2: U_p é a posição do usuário
- 3: **if** $N_c = 1$ **then**
- 4: informe ao usuário esta localização
- 5: **end if**
- 6: **if** $N_c > 1$ **then**
- 7: Calcule a distância do usuário para cada uma
- 8: **if** Suas distâncias são diferentes **then**
- 9: informe ao usuário a faixa de pedestres mais próxima
- 10: **else**
- 11: informe ao usuário a distância das duas faixas de pedestres
- 12: **end if**
- 13: **end if**

O App desenvolvido como protótipo fornece *feedback* por voz para o usuário. A fim de simplificar o retorno, considera-se que o usuário está sempre virado para a esquina ou de frente para a rua, como mencionado anteriormente. Desta forma, independentemente da posição do usuário em relação aos pontos cardeais, o *feedback* é fornecido em relação à sua direita e/ou à sua esquerda.

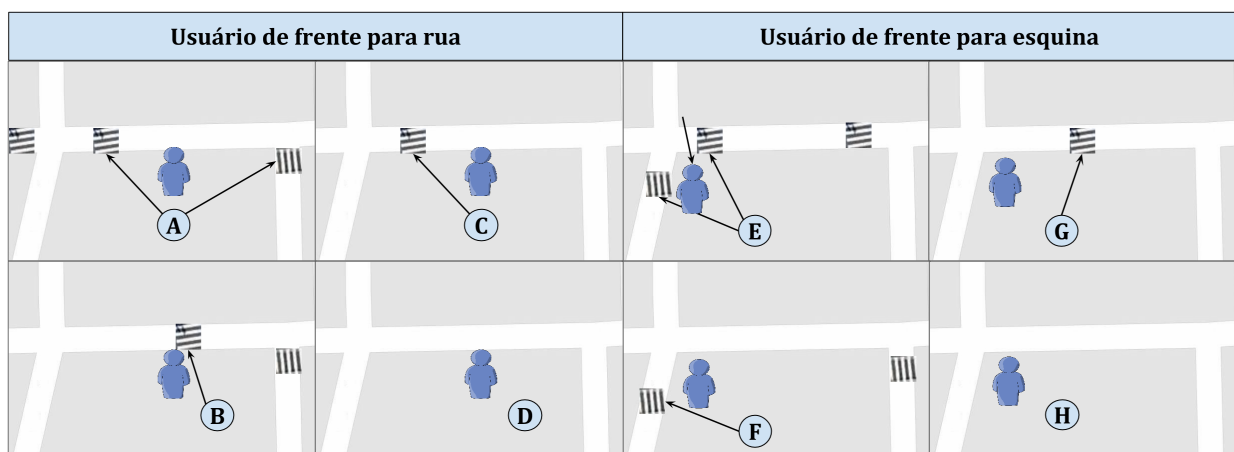


Figura 4.21 – Exemplo de possíveis *feedbacks*.

O *feedback* fornecido informa a distância até a faixa de pedestres mais próxima (ou faixas de pedestres) na quadra do usuário. A figura 4.21 exemplifica as posições do usuário e faixas que podem gerar *feedback*, e Tabela 4.2 descreve esses possíveis *feedbacks*.

Tabela 4.2 – Exemplo de *feedbacks*, ilustrados na figura 4.21.

A	Há duas faixas de pedestres em 10 metros, uma à sua esquerda e outra à sua direita, logo após virando a esquina.
B	Há uma faixa de pedestres na frente de você.
C	Há uma faixa de pedestres em 10 metros à sua esquerda.
D	Não há faixa de pedestres na quadra em que você está.
E	Você está na esquina e há duas faixas de pedestres em 4 metros, uma à sua esquerda e outra à sua direita.
F	Você está na esquina e há uma faixa de pedestres em 5 metros à sua esquerda.
G	Você está na esquina e há uma faixa de pedestres em 5 metros à sua direita.
H	Você está na esquina e não há faixas de pedestres na quadra em que você está.

Os valores apresentados na Tabela 4.2, isto é, “10 metros” e “5 metros”, são utilizadas para fins de ilustração, uma vez que as distâncias computadas (em metros) podem variar de acordo com cada situação. A informação sobre “virando a esquina” é fornecida apenas quando o usuário está em frente a linha de rua (ou seja, no meio da quadra) e a faixa de pedestre mais próxima está posicionada depois de virar a esquina.

Resultados sobre a pesquisa de localização de faixas de pedestres e obtenção de distâncias das esquinas estão descritos no capítulo 5

5. RESULTADOS OBTIDOS

Os principais algoritmos e processamentos do modelo proposto no capítulo 3 estão finalizados, mas o protótipo ainda necessita alguns refinamentos e a completa integração entre as partes que o compõe. Para avaliar a detecção de pisos táteis, e a detecção e localização das faixas de pedestres, foram elaborados e feitos testes com um *ground truth*. Neste caso, foram realizados cálculos de sensibilidade, especificidade e acurácia, cujas as fórmulas estão definidas nas Equações 5.1, 5.2 e 5.3, respectivamente,

$$\text{Sensibilidade} = \frac{TP}{TP + FN}, \quad (5.1)$$

$$\text{Especificidade} = \frac{TN}{TN + FP}, \quad (5.2)$$

$$\text{Acurácia} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5.3)$$

onde TP, TN, FP, FN e referem-se a verdadeiro positivo, verdadeiro negativo, falso positivo e falso negativo, respectivamente.

Os resultados destes testes com voluntários para detecção de pisos táteis e demais avaliações são apresentados nas seções a seguir.

5.1 Detecção de Pisos Táteis e Obstáculos

Nesta seção é apresentada a avaliação do algoritmo de detecção de pisos táteis e o resultado dos testes realizados com voluntários utilizando o protótipo de detecção de pisos táteis e obstáculos terrestres. O algoritmo de detecção de obstáculos aéreos foi testado apenas em laboratório, pois como o equipamento apresentou falhas quando usado ao ar livre, ele não pode ser incluído nos testes com os voluntários.

5.1.1 Avaliação do Algoritmo de Pisos Táteis

O protótipo para detecção de pisos táteis foi implementado na linguagem de programação C++, utilizando a biblioteca OpenCV¹ versão 3.1 e a biblioteca OpenAL² para o

¹<http://opencv.org/>

²<http://www.openal.org/>

feedback de som. Para avaliação do algoritmo implementado, foram realizados experimentos em um notebook com um processador Intel Core i5 e 4GB de memória.

Inicialmente, um *ground truth* foi criado utilizando imagens de diversas calçadas em diferentes condições de iluminação. As imagens foram adquiridas por diferentes *smartphones*, posicionados em torno de 1 metro de altura e com 45° de ângulo em relação ao chão.

Um total de 521 imagens compõem este *ground truth*, sendo que 320 são de calçadas com piso tátil e 201 são de calçadas sem piso tátil. Então para cada imagem original, existe uma imagem no *ground truth* com as bordas do piso tátil, quando presente, delimitadas por duas linhas vermelhas, como ilustra a figura 5.1. O algoritmo de avaliação recebe as linhas da imagem do *ground truth* e compara com as linhas geradas pelo algoritmo de detecção de pisos táteis para determinar a acurácia.

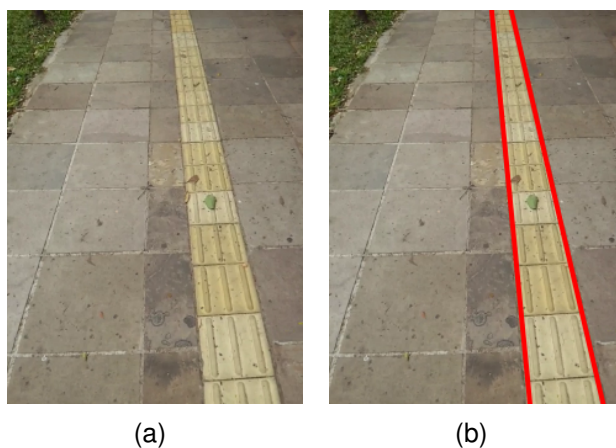


Figura 5.1 – Exemplo de *ground truth* para pisos táteis: (a) imagem original; (b) imagem do *ground truth*.

Os resultados quantitativos da avaliação estão mostrados na tabela 5.1. Os números associados com TP e TN correspondem, respectivamente, a imagens com e sem piso tátil que foram corretamente identificadas. FN e FP correspondem, respectivamente, a imagens com e sem piso tátil que foram erroneamente previstas pelo algoritmo.

Tabela 5.1 – Resultados da Avaliação

		Ground Truth	
		Piso Tátil	Sem tátil
Previsto pelo modelo	Piso Tátil	269 (TP)	47 (FP)
	Sem tátil	13 (FN)	192 (TN)

Os resultados de detecção de pisos táteis são apresentados na tabela 5.2. A especificidade de 95,52% indica que imagens sem piso tátil são previstas corretamente na maioria dos casos. O resultado sensibilidade de 84,06% e a acurácia de 88,48% demonstram a eficiência global da abordagem apresentada.

Tabela 5.2 – Avaliação de detecção de pisos táteis

Acurácia	88,48
Sensibilidade	84,06
Especificidade	95,52

5.1.2 Testes com voluntários

Foram realizados testes com voluntários para a detecção de pisos táteis e obstáculos terrestres. Através destes testes foi possível obter as percepções dos voluntários em relação a utilização desta parte do modelo.

Também seriam realizados testes do protótipo de detecção de obstáculos aéreos com os voluntários, porém não foi possível devido a falhas do equipamento durante os testes em um ambiente ao ar livre. Portanto, o algoritmo de obstáculos aéreos foi testado apenas em laboratório.

O equipamento utilizado para testar a detecção de piso tátil e de obstáculos terrestres é composto por um notebook, uma *webcam* e um fone de ouvido. Foi planejado para que o voluntário pudesse ficar com as mãos livres: a câmera foi presa na cintura do voluntário com um cinto, o fone de ouvido na cabeça sobre os ouvidos e o notebook em uma mochila colocada nas costas. Os testes foram realizados em uma calçada com aproximadamente 10 metros que continha piso tátil e obstáculos. Porém, como a bateria do notebook não suportava mais de um teste, para alguns voluntários os testes foram realizados com o auxílio de uma pessoa segurando o notebook ligado a um cabo de extensão elétrica, no lugar da mochila. Algumas imagens destes testes podem ser vista na Figura 5.2.

Cada voluntário deveria realizar as seguintes tarefas: (a) ir e voltar pela calçada no trecho pré-determinado, com os olhos vendados, apenas com a bengala padrão utilizada pelos deficientes visuais; (b) ir e voltar pela calçada no trecho pré-determinado, com os olhos vendados, apenas com o equipamento do protótipo.

Antes de iniciar os testes, cada voluntário lia e assinava um termo de concordância (Anexo A) para então ser orientado sobre as tarefas. Depois de realizar estas tarefas, ele deveria preencher um questionário pós-teste (Anexo B) para expressar suas percepções em relação ao teste e ao protótipo.

O teste foi realizado com 6 voluntários, todos do sexo masculinos e maiores de 21 anos. Dois voluntários realizaram primeiro percurso com o equipamento e depois com a bengala. Os demais começaram fizeram primeiro percurso com a bengala e depois com o equipamento. Um dos voluntários realizou o percurso que seria apenas com o equipamento utilizando também a bengala, alegando problemas de vertigem.

Como pode ser visto no Anexo B, o questionário continha perguntas referentes ao perfil do voluntário e perguntas que utilizavam a escala Likert[28], além de justificativas das

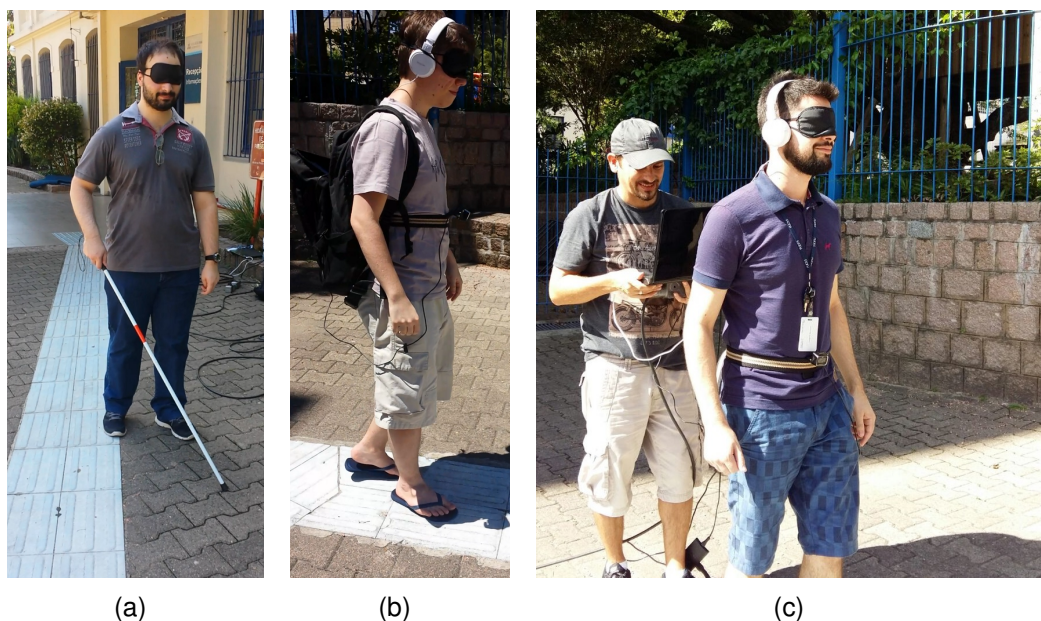


Figura 5.2 – Imagens dos testes com os voluntários: (a) apenas com o uso da bengala; (b) "vestido" com o equipamento; (c) com uma pessoa carregando parte do equipamento.

respostas e comentários. As questões mais interessantes para a avaliação eram a 6, a 8 e a 10, que utilizavam a escala Likert e são apresentadas a seguir:

Pergunta 6 - O percurso com equipamento dá maior sensação de segurança do que o percurso apenas com a bengala.

Pergunta 8 - O percurso com equipamento foi mais fácil de executar do que o percurso apenas com a bengala.

Pergunta 10 - A forma de *feedback* para o usuário é intuitiva/agradável.

As respostas para estas perguntas são apresentadas na tabela 5.3. Como se pode observar, 83% dos voluntários concordam que o equipamento proporciona maior sensação de segurança, 66% concordam que executar o percurso com o equipamento foi mais fácil do que apenas com a bengala e 83% acharam o *feedback* intuitivo/agradável.

Tabela 5.3 – Respostas dos testes dos voluntários

Voluntário	Pergunta 6	Pergunta 8	Pergunta 10
1	Não concorda parcial.	Indiferente	Concorda totalmente
2	Concorda parcialmente	Concorda parcialmente	Concorda parcialmente
3	Concorda totalmente	Concorda totalmente	Concorda parcialmente
4	Concorda parcialmente	Concorda parcialmente	Concorda parcialmente
5	Concorda parcialmente	Indiferente	Concorda parcialmente
6	Concorda parcialmente	Concorda totalmente	Não concorda parcial.

Para cada resposta, o voluntário incluía uma justificativa e comentários, e os considerados mais relevantes são apresentados a seguir:

"Som de alerta se confunde com os sons de direita e esquerda"

"Seria mais intuitivo o uso de vozes humanas no feedback de alerta e direção"

"Precisa de feedback mais claro"

"O sistema sozinho não é aconselhável mas um excelente adendo"

"Usar a bengala e o equipamento em conjunto deve ser considerado"

Nestas justificativas, há dois pontos que devem ser considerados: primeiro quanto ao *feedback*, pois verificou-se, inclusive por observação, que o *feedback* por sinais sonoros necessita de um treinamento prévio para que o voluntário se adapte aos sons. Antes do início de cada teste, os sons foram demonstrados, mas não houve um treinamento com o protótipo antes da sua efetiva utilização. Aparentemente os resultados foram promissores, mas deve-se considerar que outras formas de *feedbacks* podem ser avaliadas. Como foi mostrado no capítulo 2, há várias pesquisas específicas de *feedback* para pessoas com deficiência visual.

Outro ponto importante é quanto a utilização do protótipo. Em geral, os voluntários consideraram que o uso do equipamento torna o percurso mais fácil e com maior sensação de segurança, todavia, propuseram que o equipamento devia ser usado em conjunto com a bengala. Por observação, verificou-se que o uso da bengala deixa o voluntário mais seguro quanto a obstáculos terrestres. Algumas pesquisas demonstradas no capítulo 2 também incorporaram seus protótipos de detecção na bengala.

Observou-se também que apenas um voluntário conseguiu completar o percurso de piso tátil utilizando apenas com a bengala, mas a maioria completou o mesmo percurso com o uso do equipamento. Este resultado indica que a curva de aprendizagem para o uso de bengala é maior do que para o uso do protótipo.

Também foi possível avaliar o comportamento do protótipo durante os testes, verificando-se que a detecção de piso tátil ocorreu satisfatoriamente tanto para pisos que estavam à sombra, quanto para pisos que estavam sob o sol. Porém, na transição entre a sombra e o Sol as imagens são ajustadas automaticamente pela câmera, o que faz com que uma parte do piso tátil fique esmaecido, comprometendo, assim, a detecção do piso tátil. Nesta questão, acredita-se que a câmera e suas configurações deverão ser melhor avaliadas.

Segundo considerações dos voluntários, o protótipo auxilia e aumenta a sensação de segurança para completar a tarefa proposta, o que demonstra ser um resultado positivo para o modelo proposto e desenvolvido.

5.2 Detecção e Localização de Faixas de Pedestres

A avaliação do modelo de localização de faixas de pedestres se divide em 3 estudos de caso: (i) avaliação de extração de características, usada para escolher o melhor

método de extração de características para análise de textura (dentre as avaliadas); (ii) detecção de faixas de pedestres, utilizado para avaliar a acurácia do classificador SVM; (iii) localização de faixa de pedestres, que é usado para avaliar a acurácia do *feedback* do aplicativo.

Foram avaliados diferentes métodos de extração de características de textura (como mencionado na tabela 5.4), bem como alguma combinação deles. Os resultados experimentais, a seguir, descrevem porque LBP foi escolhido como método de extração de características.

Tabela 5.4 – Métodos para extração de características avaliadas.

Método de Extração de Características	Descritor
Gray Level Co-occurrence Matrix	Contrast Energy Entropy Homogeneity
Local Binary Pattern	Black & White Symmetry Geometric Symmetry Degree of Direction

Inicialmente, foi criado um banco de dados de imagem que contém um total de 900 imagens, é composto por 370 blocos de imagens de padrão de zebra (amostras positivas) e 530 blocos de imagens de padrão não zebra (amostras negativas), este banco de dados foi utilizado para treinar um classificador SVM para detectar as faixas de pedestres (padrão de “zebra”), como demonstrado no capítulo 4 seção 4.5 e exemplificado na figura 4.18.

Na avaliação do método de extração de características GLCM, três variáveis devem ser consideradas: a direção, a distância do vizinho e a escala de cinza (o número de tons de cinza). Uma vez que as faixas de pedestres podem assumir diferentes direções em imagens de satélite, foi realizado o experimento usando 4 direções (0° , 45° , 90° e 135°), variando o parâmetro de distância a partir de 1 até 4, com escala de cinza de 2^1 até 2^7 . Como ilustrado na figura 5.3, o melhor resultado (90,3% de acurácia) foi obtido usando distância=1 e escala de cinza=2.

Por outro lado, como pode-se ver na tabela 5.5, o melhor resultado global foi obtido pelo método de LBP, atingindo cerca de 94,6% de acurácia na detecção de faixa de pedestres. No método LBP, foram considerados uma vizinhança de 3×3 pixels, bem como as três medidas seguintes: BWS (*Black & White Symmetry*), que mede a simetria entre a metade esquerda e metade direita do histograma de escala de cinza; GS (*Geometric Symmetry*), que mede a regularidade da forma da textura; e DD (*Degree of Direction*), que mede o grau de linearidade da textura. As outras medidas geralmente utilizadas pelo método LBP não foram considerados porque elas estão relacionados com a extração de direção de textura (e as faixas de pedestres podem assumir diferentes orientações).

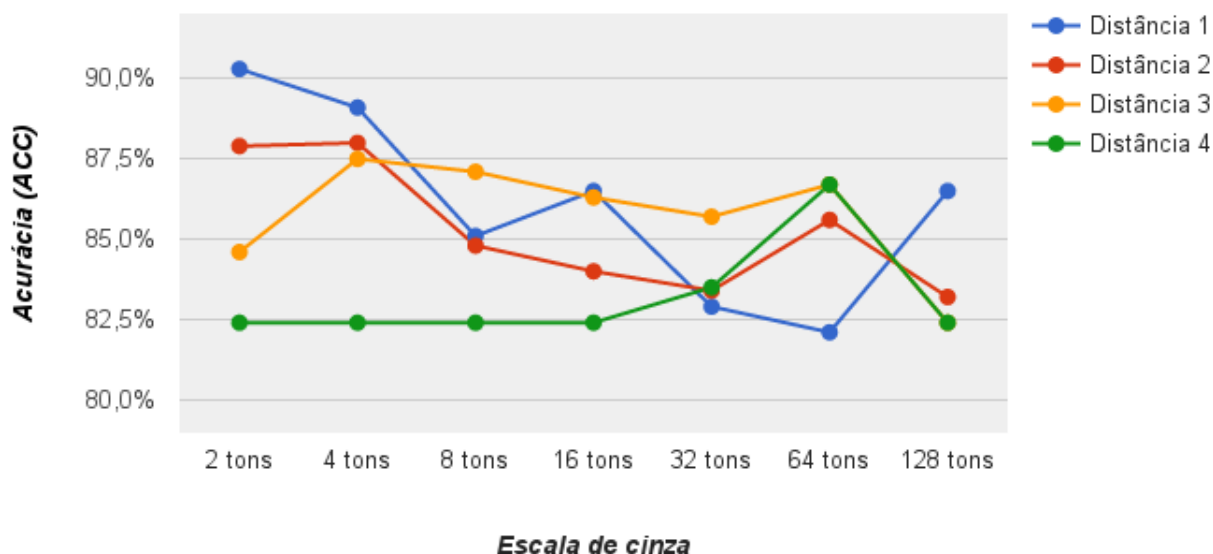


Figura 5.3 – Avaliação de acurácia do método GLCM.

Tabela 5.5 – Comparação entre métodos de extração de características.

Método	Sensibilidade	Especificidade	Acurácia
LBP	95,7%	93,9%	94,6%
GLCM + LBP	86,8%	93,6%	90,9%
GLCM	89,0%	91,1%	90,3%

Também foi avaliada a acurácia na detecção de faixas de pedestres utilizando as imagens de satélite extraídas a partir das coordenadas geográficas de um usuário. Para simular as coordenadas geográficas informadas pelo usuário foram selecionados aleatoriamente N (em que $N = 100$) coordenadas utilizando o aplicativo Google Maps (todas as coordenadas foram extraídas a partir de locais próximos de ruas, simulando a condição de que o usuário está em frente a rua ou esquina). A partir destas N coordenadas, foram extraídas as imagens pelo Google Maps API. Para cada imagem de satélite extraída um *ground truth* foi gerado manualmente. O *ground truth* (Figura 5.4(b)) de cada imagem de satélite (Figura 5.4(a)) é definido como uma imagem binária (com o mesmo tamanho) com as regiões de faixas de pedestres representadas por caixas retangulares. Estas informações são utilizadas para avaliação quantitativa. A figura 5.4(c) ilustra a saída de uma imagem binária do modelo proposto para uma dada imagem de satélite.

É importante mencionar que, neste experimento, as imagens foram extraídas de algumas capitais do Brasil, como São Paulo (SP), Florianópolis (SC) e Porto Alegre (RS). Esta escolha traz alguns desafios, ou seja, como o Brasil é considerado um país em desen-

volvimento, a qualidade de algumas faixas de pedestres pode apresentar algumas variações em relação à qualidade da pintura.

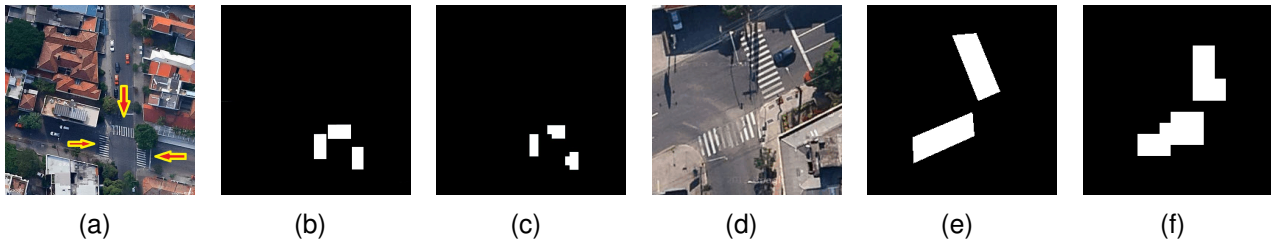


Figura 5.4 – Ilustração da imagem de satélite em (a) e um caso de faixa mal pintada (ampliado para fins de visualização) em (d) seus respectivos *ground truth* em (b) e (e). As imagens (c) e (f) ilustram as respectivas saídas de imagens binárias de o modelo proposto para a detecção de faixas.

Cada imagem do *ground truth* (Figura 5.4(b)) é comparada com a imagem de saída do modelo proposto (Figura 5.4(c)). A comparação é feita no nível de blocos em vez de pixels, as imagens são separadas em blocos de tamanho 30×30 pixels. É definido como TP quando o bloco do *ground truth* e o bloco da imagem binária na mesma localização estão marcados como faixa em pelo menos 10% de suas áreas. Um FP é definido quando o bloco da imagem binária possui faixa em mais de 10% de sua área e o bloco correspondente do *ground truth* não possui faixa. TN e FN são definidos de forma semelhante.

Como mostrado na tabela 5.6, o algoritmo desenvolvido conseguiu uma média de acurácia de 96,9%, com um desvio padrão de 2,841, o que consiste em uma taxa de acurácia satisfatória. O tempo médio para processar cada imagem foi 497 milissegundos, com desvio padrão de 244, usando um processador Intel i5, 2,27GHz and 4Gb of memory (implementação em C#.NET framework 4.5 e AForge.NET framework).

Tabela 5.6 – Avaliação de detecção de faixas.

Sensibilidade	Especificidade	Acurácia
87,5%	97,8%	96,9%

Por fim, foi avaliada a localização de faixas de pedestres. Para isto, escolheu-se aleatoriamente um conjunto de 100 coordenadas geográficas (simulando a entrada do usuário), que representam um maior número de situações, ou seja, áreas com faixa de pedestres em frente ao “usuário”, à sua direita/esquerda e em ambos os lados, bem como áreas sem faixas de pedestres e áreas com faixas de pedestres mal pintados ou com oclusões parciais. Nesta experiência, as coordenadas foram sempre obtidos a partir de uma região de calçada, em que 75% delas foram extraídas simulando que o usuário está posicionado de frente para a linha da rua enquanto que os outros 25% simulam que o usuário está posicionado na esquina.

Em uma segunda etapa, criou-se uma tabela associada ao *feedback* esperado para cada imagem extraída (por inspeção visual e de acordo com a tabela 4.2). Neste caso, considera-se um TP quando o modelo retorna o *feedback* correspondente e um FN o contrário. É importante ressaltar que a distância relativa, medida a partir do usuário para a faixa de pedestres mais próxima, é atenuada neste experimento (isso significa que pequenas variações são permitidas se as indicações correspondentes combinarem). Além disso, se existirem duas faixas a mesma distância, e o *feedback* for de apenas uma, isso é considerado um resultado falso.

Como mostrado na tabela 5.7, o modelo proposto obteve o *feedback* esperado em 92,7% dos casos simulados com uma especificidade de 95%. Considera-se estes resultados muito promissores, uma vez que incluem a localização adequada das esquinas e faixas de pedestres. valores de acurácia menores foram observados em situações em que faixas de pedestres foram mal pintadas ou sob oclusões parciais.

Tabela 5.7 – Avaliação de localização de faixas.

Sensibilidade	Especificidade	Acurácia
91,5%	95,0%	92,7%

6. CONCLUSÃO

Neste trabalho foi proposto um modelo, baseado em técnicas de visão computacional, para auxiliar pessoas com deficiência visual na navegação em calçadas. Este modelo é composto por quatro partes: detecção de piso tátil, detecção de obstáculos terrestres, detecção de obstáculos aéreos e localização de faixas de pedestres. Os dados de entrada processados no modelo são de diferentes fontes, como imagens obtidas por *webcam*, sensor ultrassônico, imagens de satélite e imagens de mapa rodoviário. Várias técnicas de processamento de imagens e aprendizagem de máquina foram utilizadas, tais como operações morfológicas, segmentação, SVM e árvores de decisão. O modelo proposto possui um baixo custo computacional e requer uma intervenção mínima do usuário.

O modelo foi avaliado quantitativamente para a detecção e localização de faixas de pedestres e detecção de pisos táteis. Para isto, foram gerados alguns conjuntos de dados como *Ground Truth*, incluindo imagens de calçadas com e sem piso tátil, imagens de satélite com e sem faixas de pedestres e lista de coordenadas geográficas com posicionamento próximo a faixas de pedestres.

Em relação à detecção e localização de faixas de pedestres, os resultados experimentais indicaram que o modelo detecta faixas de pedestres em 96,9% dos casos, atingindo cerca de 92,7% de acurácia em relação à sua localização. Um artigo intitulado "*Crosswalk localization from low resolution satellite images to assist visually impaired people*", e que contém a descrição desta parte do modelo, foi aceito para publicação em 2016 no periódico IEEE Computer Graphics and Applications. Na detecção de pisos táteis, os resultados indicaram que o modelo obteve sensibilidade de 84,06% e acurácia de 88,48% e também foram publicados em um artigo intitulado "*A new Approach for Automatic Detection of Tactile Paving Surfaces in Sidewalks*". Este artigo foi aceito para a International Conference on Computational Science – ICCS 2016. Pode-se, então, afirmar que estes resultados demonstram a eficiência global da abordagem apresentada.

Também foram realizados testes com voluntários, onde 83% dos voluntários concordam que o equipamento proporciona maior sensação de segurança, 66% concordam que executar o percurso com o equipamento foi mais fácil do que apenas com a bengala e 83% acharam o *feedback* intuitivo/agradável.

Considerando estes resultados, o modelo proposto, juntamente com os algoritmos desenvolvidos, pode ser um meio de auxiliar as pessoas com deficiência visual, ajudando-os a se locomover em ambientes ao ar livre de uma forma mais fácil e segura.

Como trabalho futuro, pretende-se melhorar a detecção de obstáculos aéreos e terrestres e a integração entre as partes que compõe o modelo, como fazer novas pesquisas que busquem resolver algumas limitações do modelo. Estas pesquisas incluem: estender a capacidade do algoritmo de detecção de pisos táteis para identificar também pisos de cores

e formatos diferentes; verificar o uso de câmara com infravermelho para que a detecção de pisos táteis e de obstáculos terrestres funcione também durante o dia. Também é interessante pesquisar a utilização de outros recursos disponíveis na maioria dos *smartphones*, como a bússola e o acelerômetro, a fim de aumentar a acurácia da localização de faixas de pedestres e melhorar a detecção de pisos táteis. Além disso, é importante desenvolver um estudo de caso com as pessoas com deficiência visual para avaliar a aplicabilidade real do modelo proposto.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Aguerrevere, D.; Choudhury, M.; Barreto, A. “Portable 3d sound / sonar navigation system for blind individuals”. In: 2nd LACCEI Int. Latin Amer. Caribbean Conf. Eng. Technol. Miami, 2004, pp. 1–10.
- [2] Araujo, T. “Sensor ultrassonico com arduino”. Blog Fazedores, 2014, Capturado em: <http://blog.fazedores.com/sensor-ultrassonico-com-arduino/>.
- [3] Barlow, J. M.; Bentzen, B. L.; Bond, T. “Blind pedestrians and the changing technology and geometry of signalized intersections: safety, orientation, and independence”, *Journal of visual impairment & blindness*, vol. 99–10, 2005, pp. 587–598.
- [4] Bourbakis, N.; Makrogiannis, S.; Dakopoulos, D. “A system-prototype representing 3d space via alternative-sensing for visually impaired navigation”, *IEEE Sensors Journal*, vol. 13–7, July 2013, pp. 2535–2547.
- [5] Bourbakis, N.; Ravraki, D. “Intelligent assistants for handicapped people’s independence: case study”. In: IEEE International Joint Symposia on Intelligence and Systems, 1996, 1996, pp. 337–344.
- [6] Bousbia-Salah, M.; Redjati, A.; Fezari, M.; Bettayeb, M. “An ultrasonic navigation system for blind people”. In: IEEE International Conference on Signal Processing and Communications, 2007. ICSPC 2007, 2007, pp. 1003–1006.
- [7] Bouzit, M.; Chaibi, A.; Laurentis, K. J. D.; Mavroidis, C. “Tactile feedback navigation handle for the visually impaired”. In: ASME Int. Mech. Eng. Congr. RD&D Expo, 2004, pp. 1–10.
- [8] Dakopoulos, D.; Boddhu, S.; Bourbakis, N. “A 2d vibration array as an assistive device for visually impaired”. In: Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering, 2007. BIBE, 2007, pp. 930–937.
- [9] Dakopoulos, D.; Bourbakis, N. “Preserving visual information in low resolution images during navigation of visually impaired”. In: Proceedings of the 1st International Conference on Pervasive Technologies Related to Assistive Environments, 2008, pp. 27:1–27:6.
- [10] D’Atri, E.; Medaglia, C.; Serbanati, A.; Ceipidor, U. “A system to aid blind people in the mobility: A usability test and its results”. In: Second International Conference on Systems, 2007. ICONS ’07, 2007, pp. 35–35.

- [11] Dixit, A.; Hegde, N. "Image texture analysis - survey". In: Third International Conference on Advanced Computing and Communication Technologies (ACCT), 2013, 2013, pp. 69–76.
- [12] Ethos, I.; colaboradores. "O que as empresas podem fazer pela inclusão das pessoas com deficiência". Instituto Ethos, 2002, Capturado em: <http://www3.ethos.org.br/>.
- [13] Faceli, K.; Lorena, A. C.; Gama, J.; Carvalho, A. "Inteligência artificial: Uma abordagem de aprendizado de máquina", *Livros Técnicos e Científicos*, vol. 1, 2011.
- [14] Fallah, N.; Apostolopoulos, I.; Bekris, K.; Folmer, E. "The user as a sensor: Navigating users with visual impairments in indoor spaces using tactile landmarks". In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2012, pp. 425–432.
- [15] Filho, J. d. S. R.; de Barros Araújo, R.; Bastos, B. L. "Projeto Argos - Auxílio à locomoção de deficientes visuais a partir de dispositivos microcontrolado". SBPC - Sociedade Brasileira para o Progresso da Ciência, 2010, 62ª reunião anual da sbpc ed., Capturado em: <http://www.sbpnet.org.br/livro/62ra/resumos/resumos/1029.htm>.
- [16] GFB. "Decreto Nº 5.296 de 2.12.2004". Governo Federal do Brasil, 2004, Capturado em: http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2004/decreto/d5296.htm.
- [17] GFB. "Decreto Nº 6.949 de 25.08.2009". Governo Federal do Brasil, 2009, Capturado em: <http://www.pessoacomdeficiencia.gov.br/app/legislacao-0>.
- [18] Hub, A.; Diepstraten, J.; Ertl, T. "Design and development of an indoor navigation and object identification system for the blind". In: Proceedings of the 6th International ACM SIGACCESS Conference on Computers and Accessibility, 2004, pp. 147–152.
- [19] IAPB. "World Blind Union". The International Agency for the Prevention of Blindness, 2014, Capturado em: <http://www.iapb.org/>.
- [20] IBGE. "Censo 2010 - Pessoas com Deficiência". Instituto Brasileiro de Geografia e Estatística, 2010, Capturado em: <http://www.ibge.gov.br>.
- [21] Ifukube, T.; Sasaki, T.; Peng, C. "A blind mobility aid modeled after echolocation of bats", *IEEE Transactions on Biomedical Engineering*, vol. 38–5, May 1991, pp. 461–465.
- [22] Ito, K.; Okamoto, M.; Akita, J.; Ono, T.; Gyobu, I.; Takagi, T.; Hoshi, T.; Mishima, Y. "Cyarm: An alternative aid device for blind persons". In: CHI '05 Extended Abstracts on Human Factors in Computing Systems, 2005, pp. 1483–1488.
- [23] Ivanchenko, V.; Coughlan, J.; Shen, H. "Detecting and locating crosswalks using a camera phone". In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08, 2008, pp. 1–8.

- [24] Jie, X.; Xiaochi, W.; Zhigang, F. "Research and implementation of blind sidewalk detection in portable eta system". In: International Forum on Information Technology and Applications (IFITA), 2010, 2010, pp. 431–434.
- [25] Johnson, L.; Higgins, C. "A navigation aid for the blind using tactile-visual sensory substitution". In: 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2006. EMBS '06, 2006, pp. 6289–6292.
- [26] Kassim, A.; Jaafar, H.; Azam, M.; Abas, N.; Yasuno, T. "Design and development of navigation system by using rfid technology". In: IEEE 3rd International Conference on System Engineering and Technology (ICSET), 2013, 2013, pp. 258–262.
- [27] Kumar, M. R. B.; Siby, C. M. "Design for visually impaired to work at industry using rfid technology". In: International Conference on Electronic Design, Computer Networks Automated Verification (EDCAV), 2015, 2015, pp. 29–33.
- [28] Likert, R. "The method of constructing and attitude scale", *Methods and Techniques in Business Research*, vol. 1, 1967, pp. 54.
- [29] Liu, J.; Sun, X. "A survey of vision aids for the blind". In: The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006., 2006, pp. 4312–4316.
- [30] Lykawka, C. "Desenvolvimento de interface de apoio à movimentação de deficientes visuais com base na representação tátil de informação de profundidade da cena", Mestrado em ciências da computação, PUCRS - FACIN - PPGCC, 2013, 89p.
- [31] Maciel, S. F. "O Ir e Vir do Deficiente Visual. Princípios, técnicas e procedimentos". CMDV - Portal do Deficiente Visual, 2003, Capturado em: http://www.cmdv.com.br/lermais_materias.php?cd_materias=568.
- [32] Meers, S.; Ward, K. "A substitute vision system for providing 3d perception and gps navigation via electro-tactile stimulation". In: Proceedings of the International Conference on Sensing Technology, Palmerston North, New Zealand, 2005, pp. 551–556.
- [33] Pedrini, H.; Schwartz, W. "Análise de imagens digitais: princípios, algoritmos e aplicações". Thomson Pioneira, 2008, 506p.
- [34] Radvanyi, M.; Paziienza, G.; Karacs, K. "Crosswalks recognition through cnns for the bionic camera: Manual vs. automatic design." In: European Conference on Circuit Theory and Design, 2009. ECCTD 2009., 2009, pp. 315–318.
- [35] Sainarayanan, G.; Nagarajan, R.; Yaacob, S. "Fuzzy image processing scheme for autonomous navigation of human blind", *Applied Soft Computing*, vol. 7–1, 2007, pp. 257 – 264.

- [36] Shah, C.; Bouzit, M.; Youssef, M.; Vasquez, L. "Evaluation of ru-netra - tactile feedback navigation system for the visually impaired". In: International Workshop on Virtual Rehabilitation, 2006, 2006, pp. 72–77.
- [37] Shoval, S.; Borenstein, J.; Koren, Y. "Mobile robot obstacle avoidance in a computerized travel aid for the blind". In: Proceedings of the IEEE International Conference on Robotics and Automation, 1994, 1994, pp. 2023–2028 vol.3.
- [38] Simoes, W. C. S. S.; de Lucena, V. F. "Blind user wearable audio assistance for indoor navigation based on visual markers and ultrasonic obstacle detection". In: IEEE International Conference on Consumer Electronics (ICCE), 2016, pp. 60–63.
- [39] Sousa, K.; Marengoni, M. "Uso de visão computacional em dispositivos moveis para o reconhecimento de faixa de pedestres". In: Anais do VIII Workshop de Visão Computacional, 2012, pp. 1–85.
- [40] Tudor, D.; Dobrescu, L.; Dobrescu, D. "Ultrasonic electronic system for blind people navigation". In: E-Health and Bioengineering Conference (EHB), 2015, 2015, pp. 1–4.
- [41] Ulrich, I.; Borenstein, J. "The guidecane-applying mobile robot technologies to assist the visually impaired", *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 31–2, Mar 2001, pp. 131–136.
- [42] Valentim, R. "Olho Bionico". Laboratório de Inovação Tecnológica na Saúde da UFRN, 2014, Capturado em: <http://f123.org/archives/4583>.
- [43] Vera, P.; Zenteno, D.; Salas, J. "A smartphone-based virtual white cane", *Pattern Analysis and Applications*, vol. 17–3, Ago 2014, pp. 623–632.
- [44] Wang, S.; Tian, Y. "Detecting stairs and pedestrian crosswalks for the blind by rgb-d camera". In: IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW), 2012, 2012, pp. 732–739.
- [45] Woo, B.-S.; Yang, S.-M.; Jo, K.-H. "Brick path detection from shape pattern and texture feature". In: IEEE/SICE International Symposium on System Integration (SII), 2011, 2011, pp. 78–83.
- [46] Woo, B.-S.; Yang, S.-M.; Vavilin, A.; Jo, K.-H. "Brickpath region detection using color and shape pattern information". In: 6th International Forum on Strategic Technology (IFOST), 2011, 2011, pp. 720–724.

ANEXO A – Termo de consentimento

Projeto modelo de apoio à navegação em calçadas para pessoas com deficiência visual

Faculdade de Informática/PUCRS
Avenida Ipiranga, 6681 – Prédio 32 - 90619-900 – Porto Alegre – RS
Tel: (51) 3320-3558

Termo de Consentimento Livre e Esclarecido

A equipe da Faculdade de Informática do **Projeto modelo de apoio à navegação em calçadas para pessoas com deficiência visual**, agradece a todos os participantes de testes realizados sob sua responsabilidade, a inestimável contribuição que prestam para o avanço da pesquisa.

O objetivo desta pesquisa é investigar métodos de auxílio à navegação em calçadas para pessoas com deficiência visual. Para isto, os participantes dos testes são convidados a usar o protótipo do modelo, enquanto são observados por um pesquisador. Esta observação será registrada em texto e, também, através de gravação de vídeo. Estas informações nos trarão dados importantíssimos para verificar o potencial do modelo em questão.

Lembramos que o objetivo deste estudo **não é** avaliar o participante, **mas sim** avaliar o protótipo que o participante utilizará durante os testes. O uso que se faz dos registros efetuados durante o teste é **estritamente** limitado a atividades de pesquisa e desenvolvimento, garantindo-se para tanto que:

1. O anonimato dos participantes será preservado em todo e qualquer documento divulgado em foros científicos (tais como conferências, periódicos, livros e semelhantes) ou pedagógicos (tais como apostilas de cursos, *slides* de apresentações, e semelhantes).
2. Todo participante terá acesso a cópias destes documentos após a publicação dos mesmos.
3. Todo participante que se sentir constrangido ou incomodado durante uma situação de teste pode interromper o teste e estará fazendo um favor à equipe se registrar por escrito as razões ou sensações que o levaram a esta atitude. A equipe fica obrigada a descartar o teste para fins da avaliação a que se destinaria.
4. Os participantes que forem menores de idade terão, obrigatoriamente, que apresentar o consentimento de seu responsável, para participação no estudo, o qual será declarado ciente do estudo a ser realizado através de sua assinatura no presente Termo de Compromisso.
5. Todo participante tem direito de expressar por escrito, na data do teste, qualquer restrição ou condição adicional que lhe pareça aplicar-se aos itens acima enumerados (1, 2, 3 e 4). A equipe se compromete a observá-las com rigor e entende que, na ausência de tal manifestação, o participante concorda que rejeita o comportamento ético da equipe somente as condições impressas no presente documento.
6. A equipe tem direito de utilizar os dados dos testes, mantidas as condições acima mencionadas, para quaisquer fins acadêmicos, pedagógicos e/ou de desenvolvimento contemplados por seus membros.

[a ser preenchido pelo observador]
Sistema: _____ Data: __/__/__
Condições especiais (caso não haja condições especiais, escreva "nenhuma"):

[] continua no verso

Por favor, indique sua posição em relação aos termos acima:

[] Estou de pleno acordo com os termos acima.

[] Em anexo registro condições adicionais para este teste.

Assinatura do participante

Assinatura do responsável
(caso o participante seja menor de idade)

Assinatura do observador

Nome do Participante:

Nome do Responsável (se o participante for menor de idade):

Pesquisador Responsável: Prof. _____ – Faculdade de Informática - PUCRS

ANEXO B – Questionário pós-teste

Avaliação do modelo de apoio à navegação em calçadas para pessoas com deficiência visual

PERFIL

1. Idade: entre 14 e 20 anos
 entre 21 e 30 anos
 mais de 31 anos

2. Sexo: Masculino Feminino

3. Escolaridade mais alta:
 Ensino Fundamental Completo
 Ensino Médio Completo
 Graduação
 Especialização
 Mestrado
 Doutorado
 Pós-Doutorado

4. Profissão: _____

MODELO

5. Qual a ordem de percurso que realizou:
 1º percurso Apenas bengala, 2º percurso com Equipamento do Projeto
 1º percurso com Equipamento do Projeto e 2º percurso Apenas bengala

6. O percurso com equipamento dá maior sensação de segurança do que o percurso apenas com a bengala.
 Não concordo totalmente
 Não concordo parcialmente
 Indiferente
 Concordo parcialmente
 Concordo totalmente

7. Se desejar, justifique a resposta colocada na questão 7:

