



UNIVERSIDADE FEDERAL DO PARÁ
PROGRAMA DE PÓS-GRADUAÇÃO DE NEUROCIÊNCIAS E
BIOLOGIA CELULAR

LEONARDO MIRANDA ABDON
Nº DE MATRÍCULA 202017470002

TECNOLOGIA DE PROCESSAMENTO DE IMAGEM PARA
ORIENTAÇÃO DE DEFICIENTES VISUAIS DENTRO DO ESPAÇO
RESIDENCIAL

BELÉM – PA
2024

LEONARDO MIRANDA ABDON
Nº DE MATRÍCULA 202017470002

**TECNOLOGIA DE PROCESSAMENTO DE IMAGEM PARA
ORIENTAÇÃO DE DEFICIENTES VISUAIS DENTRO DO ESPAÇO
RESIDENCIAL**

Dissertação de Mestrado desenvolvido sob orientação do Profº Dr. Manoel da Silva Filho, no Programa de Pós-Graduação em Neurociências e Biologia Celular (PPGNBC) da Universidade Federal do Pará (UFPA).

BELÉM – PA

2024

LEONARDO MIRANDA ABDON
Nº DE MATRÍCULA 202017470002

**TECNOLOGIA DE PROCESSAMENTO DE IMAGEM PARA
ORIENTAÇÃO DE DEFICIENTES VISUAIS DENTRO DO ESPAÇO
RESIDENCIAL**

Dissertação de Mestrado desenvolvido sob orientação do Profº Dr. Manoel da Silva Filho, no Programa de Pós-Graduação em Neurociências e Biologia Celular (PPGNBC) da Universidade Federal do Pará (UFPA).

Belém, ____ de ____ de 2024.

BANCA EXAMINADORA

_____- Orientador
Prof. Dr. Manoel da Silva Filho
Universidade Federal do Pará - UFPA

Prof. Dr. Ronaldo Nonato Ferreira Marques de Carvalho
Universidade Federal do Pará – UFPA

Prof. Dra. Cybelle Salvador Miranda
Universidade Federal do Pará – UFPA

“Um dia em que nada aprendi foi um dia em que nada vivi.”

Palavras de Leonardo da Vinci,
a mim faladas por meu avô, Raymundo Oliveira Miranda.

AGRADECIMENTOS E DEDICATÓRIAS

Primeiramente, gostaria de agradecer aos meus professores, principalmente os do curso de Arquitetura e os do curso do Mestrado em Neurociências. Sem todo o conhecimento que me passaram, mal poderia pensar em metade do que foi feito neste trabalho. E sem o suporte laboratorial, a outra metade não seria possível. Acima de tudo, gostaria de agradecer ao meu professor, orientador e amigo Manoel da Silva Filho. Muito obrigado pela paciência, pela força, pela coragem e por me inspirar a sempre ser mais do que sou hoje!

Muito obrigado aos meus amigos do LAPA: Paulo Vitor, Rafael, Gustavo, Bruna, Ana, Carol, Maria Clara e Paula! Obrigado também aos que passaram pelo laboratório e seguiram seus rumos para outros lugares!

Muitíssimo obrigado aos meus amigos queridos, os irmãos que a vida me deu: Paulo Vitor (de novo!), Leon, Letícia, Toshiaki, Akina, Allan, Gabi, Gabriel e Arthur! Obrigado aos meus amigos de profissão, os que dividiram projetos na madrugada há anos: Mariana, Fernanda, Renan, Larissa, Rafael e Inês! Às pessoas que me acompanharam de perto mesmo estando tão longe: Mário, Eduardo e Gabriela, Tassiana e todo o grupo que fez jogatinas pelo Discord! Ao pessoal da AKP, agradeço por me acolherem no kendô e iaidô, por me aceitarem como parte do cardume! E também, gostaria muito de agradecer ao meu amigo e psicólogo, Elielton, por me ajudar a aguentar esse peso todo e a me ensinar a olhar para frente com força e fé.

Obrigado a uma pessoa maravilhosa que, apesar de estar me acompanhando a pouco tempo, apareceu na minha vida como um anjo e mostrou coisas maravilhosas. Fico muito feliz de ter lhe conhecido durante o final deste ciclo da minha vida e tenho esperança de que o futuro de cada um de nós será repleto de alegria, Tatiane!

Obrigado ao meu melhor amigo, Paulo Vitor. Já o mencionei por nome três vezes só nesse texto, nem preciso dizer o quanto nossa amizade é especial para mim. Meu companheiro para, *literalmente*, todas as horas! Eu espero continuar a ter sua companhia independente do tempo ou lugar, *meu irmão*!

Por fim, muito, *muito obrigado* à minha família, que me deu suporte emocional incondicional, especialmente minha irmã Nathália, meu cunhado Henrique e minha mãe Rosalba! Muito obrigado por incentivarem meus sonhos e por sempre acreditarem no meu potencial! Por me acompanharem nas minhas noites em claro, por me motivarem pela manhã, pelas chamadas de atenção e pelos abraços, presenciais e virtuais! Obrigado! *Thank you! Merci!*

Muito obrigado a todos pelo apoio ontem, hoje e sempre! Dedico este trabalho a vocês!

RESUMO

Este trabalho visa o desenvolvimento de tecnologia assistiva baseada em plataforma Raspberry Pi com o intuito de permitir maior autonomia dentro de ambientes internos para indivíduos com cegueira. Busca-se, portanto, a montagem de um sistema vestível de detecção de obstáculos e orientação de caminhos que se utilize de câmera para captar imagens direcionadas ao piso imediatamente à frente do usuário, a fim de que se identifiquem códigos QR criados especificamente para gerar respostas dentro do algoritmo escrito, para que sejam emitidas respostas por meio de padrões vibratórios em motores e frases gravadas por fones de ouvido. O sistema foi desenvolvido desta maneira de modo a permitir autonomia à escolha de caminhos pelo usuário com auxílio de um produto de tecnologia assistiva de baixo custo, de fácil acesso e manutenção, e grande flexibilidade quanto às necessidades e preferências do usuário. O sistema teve seu funcionamento testado de maneira fixa em sala de estudo controlada e sem janelas e de maneira móvel em corredor controlado e obteve resultados satisfatórios no que se refere às distâncias de detecção de caminhos e obstáculos e à velocidade de emissão de respostas. Por fim, foi desenhado e confeccionado um suporte com alças para tornar o sistema desenvolvido num aparato vestível de maneira simples e não intrusiva.

Palavras-Chave: deficiência visual, espaço residencial, orientação, Raspberry Pi, tecnologia assistiva.

ABSTRACT

This work seeks to develop a product of assistive technology based on the Raspberry Pi platform in order to allow greater autonomy inside residential spaces for individuals with blindness. Therefore, it was assembled a wearable obstacle detection and path orienting system that uses a camera to capture images directed to the floor immediately in front of the user, in order to identify QR codes created specifically to generate responses within the written algorithm, so that responses are emitted through vibratory patterns in motors and recorded sentences through headphones. The system was developed in this way in order to allow autonomy in the choice of paths by the user with the aid of a low-cost assistive technology product, easily accessible and maintainable, and flexible in regard to the user's needs and preferences. The system's capabilities were tested in a fixed setup inside a controlled study room without windows and in a movable setup in a controlled corridor, and the results were satisfactory regarding to the detection distances of paths and obstacles and the swiftness of emission of responses. Finally, a body vest made with bag straps was designed and the developed system was installed into it, to be used as a simple and nonintrusive wearable device.

Index Terms: assistive technology, orientation, Raspberry Pi, residential spaces, visually impaired.

SUMÁRIO

1 INTRODUÇÃO	2
1.1 TÍTULO DO TRABALHO	2
1.2 DELIMITAÇÃO DO TEMA	2
1.3 JUSTIFICATIVA	2
1.4 PROBLEMA DE PESQUISA	3
1.5 OBJETIVOS:	4
1.5.1 Geral:	4
1.5.2 Específicos:	4
2 METODOLOGIA	4
2.1 HARDWARE	4
2.2 SOFTWARE	5
2.3 FUNCIONAMENTO	6
2.4 TESTES	7
2.5 ANÁLISE ESTATÍSTICA	8
3 CAPÍTULO 1: BASES TEÓRICAS	8
3.1 MOTIVAÇÕES	8
3.2 ARQUITETURA	10
3.3 TECNOLOGIA ASSISTIVA	12
3.4 COMPONENTES E CUSTOS	16
4 CAPÍTULO 2: DESENVOLVIMENTO DO ALGORITMO	19
4.1 PLANEJAMENTO INICIAL	19
4.2 MONTAGEM PRELIMINAR DO SISTEMA	22
4.3 CONSTRUÇÃO DO ALGORITMO	22
4.4 AJUSTES DE IMAGEM, PRECISÃO E DISTÂNCIA	16
4.5 MOTORES VIBRATÓRIOS E DEFINIÇÃO DE POSIÇÃO E DISTÂNCIA	28
5 CAPÍTULO 3: MONTAGEM DO PROTÓTIPO E TESTES FINAIS	31
5.1 FUNCIONAMENTO DA CÂMERA EM POSIÇÃO FIXA	31
5.2 TESTE MÓVEL EM CORREDOR	39
5.3 PROTÓTIPO	48

6 CONCLUSÃO	53
7 TRABALHOS FUTUROS	55
REFERÊNCIAS	56
APÊNDICES.....	61
APÊNDICE A – CÓDIGO EM PYTHON	61
APÊNDICE B – DETALHAMENTO DA CAIXA DE PROTEÇÃO.....	70

SUMÁRIO DE IMAGENS

Figura 1. Fluxograma do algoritmo desenvolvido. Imagem criada pelo autor.....	5
Figura 2. Esquema preliminar. Imagem criada pelo autor.	19
Figura 3. Planejamento do alcance da câmera. Imagem criada pelo autor.....	20
Figura 4. Planejamento da situação padrão. Imagem criada pelo autor.	21
Figura 5. Planejamento de detecção de um obstáculo. Imagem criada pelo autor.	21
Figura 6. Raspberry Pi 3 Model B+ utilizada (A). Webcam Logitech C270, fixada na mesa (B). Imagens registradas pelo autor.	22
Figura 7. Sinais visuais em código QR Obstacle (“obstáculo”), Way ahead (“caminho à frente”) e End (“fim”). Imagem registrada pelo autor.	22
Figura 8. Exemplo de identificação bem-sucedida de código QR e feedback da câmera em janela (A) e de identificação bem-sucedida de sinal QR rotacionado (B, C). Imagens registradas pelo autor e editadas para seletivamente borrar o fundo de modo a enfatizar o funcionamento do dispositivo.....	22
Figura 9. Feedback em resolução menor e em preto e branco gerado na webcam Logitech C270. Imagem registrada e editada pelo autor.	15
Figura 10. Exemplo de respostas emitidas perante identificação de código QR. Imagem registrada e editada pelo autor.	16
Figura 11. Respostas “precisa” (A) e “não precisa” (B). Imagem registrada pelo autor.....	17
Figura 12. Imagens coloridas (A, B, C) e em preto e branco (D, E, F) referentes às distâncias de 21 (A, D), 120 (B, E), 237 (C) e 260 (F) centímetros. Imagens registradas e editadas pelo autor.....	26
Figura 13. Situações onde os sinais “Way ahead” (A) e “Way to the left” (B) estão mais próximos da câmera em diferentes momentos. Imagens registradas e editadas pelo autor.....	27
Figura 14. Diagrama das portas GPIO das placas Raspberry. Adaptado de <i>Raspberry Pi Documentation</i> (link para acesso: < https://www.raspberrypi.com/documentation/computers/raspberry-pi.html >).	28
Figura 15. Representação visual da divisão dos setores. Imagem registrada e editada pelo autor.	29
Figura 16. Fluxograma da inicialização do algoritmo desenvolvido. Imagem criada pelo autor.	31
Figura 17. Imagens do sistema fixo, com a placa Raspberry Pi presa a um tripé (A), ligada a um monitor e alimentada por uma tomada (B). Imagens registradas pelo autor.	32

Figura 18. Detalhe da placa Raspberry Pi com case de proteção (A) e câmera fixada e rotacionada 180° (B). Imagens registradas pelo autor.	33
Figura 19. Distância máxima de identificação por altura verificada. Imagem criada pelo autor.	34
Figura 20. Taxa de identificação por altura da câmera, comparados ao controle (CT). Imagem criada pelo autor.	35
Figura 21. Simulação tridimensional do campo de visão da câmera. Imagem criada pelo autor.	35
Figura 22. Projeção da área do campo de visão baseada nas especificações da câmera. Imagem criada pelo autor.	36
Figura 23. Divisão e dimensão dos setores no campo de visão da câmera. Imagem criada pelo autor.	37
Figura 24. Taxa de identificação por setor, comparados à situação controle (CT). Imagem criada pelo autor.	38
Figura 25. Configuração para o teste de iluminação. Imagem registrada pelo autor.	38
Figura 26. Taxa de identificação por nível de iluminação, comparados à situação controle (CT). Imagem criada pelo autor.	39
Figura 27. Mesa utilizada para os testes móveis, com o sistema, monitor, teclado e mouse. Imagem registrada e editada pelo autor.	40
Figura 28. Simulação tridimensional cotada do trecho do corredor utilizado no primeiro conjunto de testes móveis. Imagem criada pelo autor.	41
Figura 29. Demonstração da capacidade apropriada de identificação do dispositivo no que se refere aos variados níveis de intensidade luminosa presentes no corredor. Imagens registrada pelo autor.	42
Figura 30. Simulação tridimensional cotada do trecho do corredor utilizado no segundo conjunto de testes móveis. Imagem criada pelo autor.	43
Figura 31. Sistema ligado na bateria, com tela para monitoramento das imagens. Imagem registrada e editada pelo autor.	43
Figura 32. Sistema identificando um código QR em situação padrão (A) e rotacionado em 180° (B). Imagens registradas pelo autor.	44
Figura 33. Sistema identificando um código QR em diferentes distâncias. Imagens registradas pelo autor.	45
Figura 34. Sistema identificando códigos QRs simultâneos. Imagens registradas pelo autor.	46

Figura 35. Teste de funcionamento do sistema sem tela e ligado à bateria, à velocidade de caminhar de aproximadamente 0,4 m/s (1,44 km/h). Imagem registrada e editada pelo autor.	47
Figura 36. Esquema atualizado, com vista frontal (A) e dorsal (B). Imagem criada pelo autor.	49
Figura 37. Imagens das alças confeccionadas para o sistema de frente (A), de costas (B), em perfil (C, D), e detalhes dos encaixes e ajustes (D). Imagens registradas pelo autor.	50
Figura 38. Dimensões finais do modelo da caixa a ser impressa. Imagem criada pelo autor.	51
Figura 39. Imagens das peças de impressão 3D em vistas diferentes (A: inferior direita. B: superior esquerda). Imagens registradas pelo autor.	51
Figura 40. Imagens do modelo de impressão 3D montado em vistas diferentes (A: inferior direita. B: superior esquerda). Imagens registradas pelo autor.	52
Figura 41. Imagens do protótipo finalizado (A, B, C). Imagens registradas pelo autor.	52

1 INTRODUÇÃO

1.1 TÍTULO DO TRABALHO

Tecnologia de processamento de imagem para orientação de deficientes visuais dentro do espaço residencial.

1.2 DELIMITAÇÃO DO TEMA

Desenvolvimento de objeto tecnológico voltado para facilitar o deslocamento de indivíduos com deficiências visuais em escala residencial.

1.3 JUSTIFICATIVA

A tecnologia na arquitetura evolui conforme o ser humano interage com o espaço construído, buscando resolver desafios diversos e recorrendo a adaptações em edifícios que não mais atendem às necessidades vigentes. Deste modo, o desenvolvimento das *smart homes* foi uma etapa natural da interação do ser humano com a alta tecnologia em seu ambiente privativo e profissional.

A convivência com ambientes altamente inteligentes é recente no cenário arquitetônico nacional, uma vez que em nosso país o custo desse tipo de construção é muito alto, tornando-se limitado a edifícios de luxo e empresariais. Outra forma de alcançar um efeito similar é a aquisição e instalação de móveis capazes de se integrar uns aos outros, via *Bluetooth*, rede de internet ou outra forma de comunicação. Porém, deixar a adequação do ambiente a cargo do comprador contrapõe a ideia do uso da alta tecnologia como parte integral do conceito do projeto. Qualquer que seja o modo, a alta tecnologia no espaço construído é um avanço quanto ao conforto e às novas interações entre humano e máquina.

Deste modo, também há de se falar em tecnologia assistiva, um ramo crescente na indústria devido à demanda por espaços acessíveis. Mesmo que os elementos assistivos evoluam e incorporem-se aos projetos arquitetônicos em ritmo acelerado, ainda é possível pontuar exemplos de edifícios mal adaptados, assim como aqueles que, apesar de adequados legalmente, são confusos, escondidos ou mesmo desconfortáveis para o uso.

1.4 PROBLEMA DE PESQUISA

O foco deste trabalho se encontra nas necessidades das pessoas com deficiências visuais. Esta escolha foi feita em razão da predominância dos elementos assistivos para essas pessoas fora da esfera particular.

Como exemplo, os pisos táteis, apesar de serem legalizados e regulamentados pelas normas ABNT NBR 9050 (2015) e ABNT NBR 16537 (2016), são majoritariamente instalados em vias de acesso público, como passeios e ambientes comerciais, portanto incomum no interior de espaços privativos. Semelhantemente, outras adaptações táteis, como placas e texturas para indicar direções e informações, são pouco usadas no espaço residencial, sendo mais encontradas em botões de elevador e placas de emergências.

Tais preocupações já são observadas em pesquisas acadêmicas em universidades estrangeiras, como vistas em TJAN et al (2005), que desenvolve um sistema de sinais digitais para indicação de ambientes, e em WACHAJA et al (2015, 2017), que abordou o desenvolvimento de um “andador inteligente” para navegação de pessoas cegas com problemas de locomoção, o estudo de uma pulseira com arranjo de múltiplos motores vibratórios de LEE e IN (2023), o sistema baseado na plataforma Nvidia Jetson de MADAKE et al (2023), assim como os trabalhos de TEPELEA et al (2019), que estuda a criação de um módulo visual para deficientes visuais dentro da plataforma Raspberry Pi, e o de KLEINBERG et al (2023), que também se utiliza da plataforma Raspberry Pi em conjunto com sensores ultrassônicos e visuais. Outros trabalhos também foram estudados a fim de esclarecer as formas alternativas como tem se usado a tecnologia para tratar o assunto, como os sistemas baseados em *smartphones* de compreensão de ambientes de JEON et al (2012), TAPU et al (2013) e de CHACCOUR e BADR (2015), entre outros estudos de estado da arte dos anos 2015 a 2023.

Embasado nessas questões, estuda-se a viabilidade de um sistema de orientação inteligente e portátil a ser utilizado em ambientes privativos voltado para pessoas com deficiências visuais, que lhes seja acessível tanto quanto ao seu uso quanto ao custo do equipamento.

1.5 OBJETIVOS:

1.5.1 Geral:

1. Desenvolver uma tecnologia vestível baseado na plataforma Raspberry, utilizando-se de processamento de imagem de sinais visuais baseados em *códigos visuais do tipo Quick Response* (códigos QR) para auxiliar a orientação independente em ambientes internos de pessoas com deficiências visuais, tipo cegueira.

1.5.2 Específicos:

1. Construir tecnologia vestível de baixo custo utilizando a plataforma Raspberry, que usará módulos de câmera e motores de resposta tátil mediante processamento de imagem baseado em uso de sinais visuais de código QR.
2. Escrever algoritmo para rodar no sistema, de forma autônoma e portátil.
3. Testar o tempo, distância e precisão de identificação e respostas emitidas do equipamento num caminho com obstáculos elaborado em ambiente laboratorial.

2 METODOLOGIA

2.1 HARDWARE

Para fazer o processamento principal, foi utilizado o *hardware* Raspberry Pi 3 Model B+, um microcomputador com sistema Raspbian, um sistema operacional baseado em Linux. Para a produção e leitura de imagens, foi utilizado um módulo de câmera próprio da placa Raspberry Pi. Para emissão de respostas ao usuário, foram utilizados dois motores de vibração do tipo moeda, modelo 1027, que se ativam mediante confirmações alcançadas pelo código de programação, definidos como “motor direito” e “motor esquerdo”. Para prover energia e tornar o projeto portátil, foi utilizado uma de bateria externa com capacidade de 9600mAh e saída de 5V e corrente de até 4,5 A. Para a sinalização visual, foram criados três sinais visuais no formato de código QR, impressos em papel A4: “C” para “Caminho”, “O” para “Obstáculo” e “F” para “Fim do Caminho”. Foi confeccionado um colete para abrigar todos os componentes de forma não-intrusiva e fácil utilização.

2.2 SOFTWARE

O intuito para o projeto foi de que trabalhasse de forma autônoma, então foi escrito um algoritmo na linguagem Python, lido pelo próprio sistema operacional da Raspberry Pi 3 no momento em que for ligada. O algoritmo foi baseado em outros fornecidos de forma gratuita na internet (código aberto ou “open-source”), porém adaptado e reescrito conforme as necessidades voltadas para o projeto em questão. Ele é responsável por ativar a câmera, obter as imagens, analisar os sinais visuais e emitir respostas correspondentes aos motores de vibração. Este algoritmo rodará em loop para cada ciclo bem-sucedido, seguindo as seguintes etapas: leitura das imagens obtidas, identificação de códigos QR cadastrados (rotacionados ou não), emissão de respostas, finalização da resposta. No caso de não haver código QR cadastrado nas imagens da câmera, o sistema continua analisando as imagens em tempo real até obter correspondência. O sistema não emite respostas para códigos QR não cadastrados. No caso de múltiplos sinais QR presentes nas imagens, apenas o mais próximo é considerado para emissão de respostas.

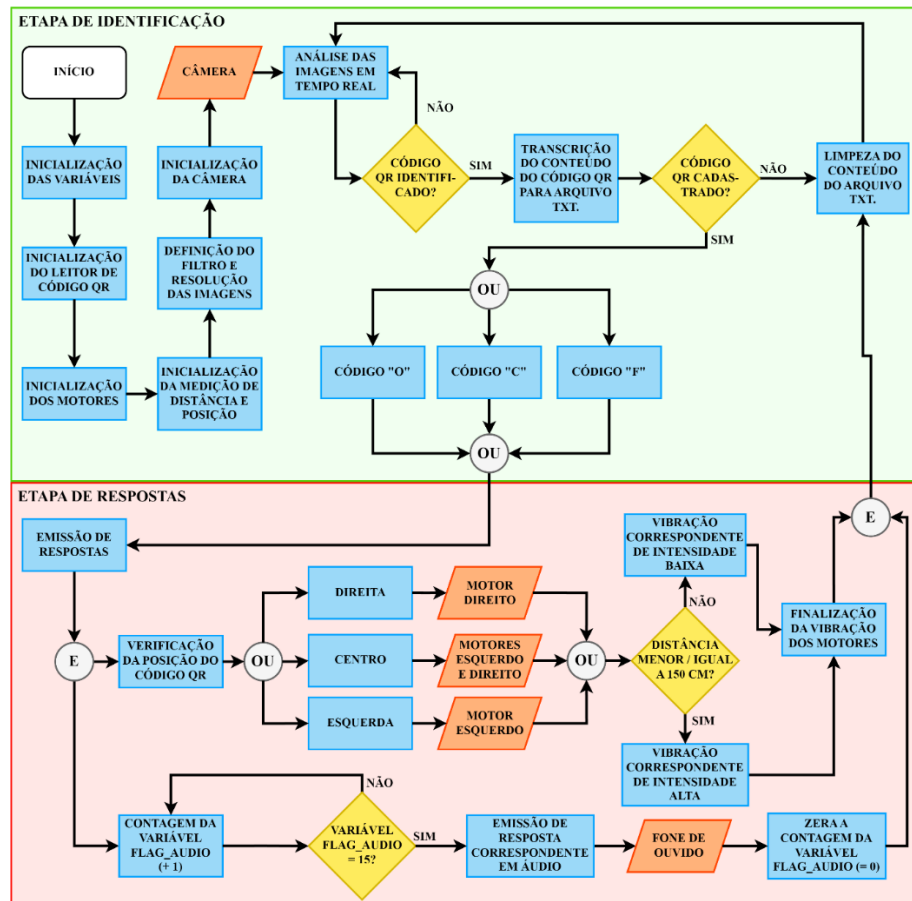


Figura 1. Fluxograma do algoritmo desenvolvido. Imagem criada pelo autor.

2.3 FUNCIONAMENTO

O sistema é vestível pelo usuário na forma de um colete. Os dois motores de vibração se posicionam nos ombros direito e esquerdo, enquanto a placa Raspberry Pi e a câmera nela instalada se localizam na parte peitoral do usuário, posicionada no centro do colete. O algoritmo lido pelo sistema operacional da Raspberry Pi é responsável por analisar as imagens da câmera em tempo real. Para a análise das imagens, considerou-se que a velocidade média de deslocamento de uma pessoa sem deficiências visuais é de 1,25 *metros por segundo* (m/s), ou 4,5 *quilômetros por hora* (km/h), como diz SCHIMPL *et al* (2011), enquanto que a velocidade média de uma pessoa com cegueira é mais baixa que pessoas com visão normal (CLARK-CARTER, 1986). Portanto, foi considerada velocidade de 0,4 m/s (1,44 km/h) a 0,7 m/s (2,5 km/h) como deslocamento para se definir o intervalo entre ciclos de aquisição de imagens 0.05 segundos, proporcionando a leitura das imagens da câmera a 10 *quadros por segundo* (QPS) durante emissão de respostas, e funcionando a 30 QPS quando não há emissão de respostas, para que o sistema responda mais rapidamente ao detectar o próximo sinal visual. A situação padrão é de que a Raspberry Pi somente acione os motores perante um reconhecimento bem-sucedido de um sinal QR. Na ausência de sinais visuais, o sistema permanece em loop. Mediante sinais encontrados nas imagens analisadas, o código roda a cada 0,1 *segundo* (s) para reduzir a carga de processamento e proporcionar tempo para respostas em frequências específicas de vibração enquanto os sinais permanecerem nas imagens da câmera. A intensidade e padrões de vibrações variam conforme identificação dos códigos visuais, portanto cada código QR emite uma resposta particular. Se os sinais visuais se encontrarem à esquerda dentro das imagens captadas, é ativado o motor de vibração esquerdo de acordo com o sinal cadastrado e a distância registrada, e o mesmo vale para situações onde os sinais visuais estiverem na direita, assim ativando o motor de vibração direito. Quando os sinais estiverem posicionados no centro das imagens, são acionados os dois motores simultaneamente. Os sinais para indicar caminhos possíveis, “C”, emitem resposta vibratória curta e intervalada. Os sinais para indicar obstáculo, “O”, emitem respostas em alta potência e duração constante nos motores. O sinal “final do caminho” ativa os motores a baixa potência e curta duração.

2.4 TESTES

Os primeiros testes foram realizados a fim de verificar a eficácia do algoritmo escrito, com a Raspberry Pi conectada à um monitor e abastecida por entrada USB 3.0 de computador. As respostas foram emitidas na forma de frases dentro do terminal de software de programação próprio da Raspberry Pi. Nesta etapa também foram realizados testes com diversos filtros nas imagens da câmera a fim de verificar qual produzia respostas mais precisas e mais rápidas. Ainda de maneira fixa, foram realizados testes para verificar as distâncias nas quais o código é efetivo e se múltiplos sinais QR poderiam afetar a detecção precisa, assim como a definição de padrões vibratórios, uma vez instalados os motores no sistema. Por fim, foi analisada a precisão do sistema, considerando um número controle de identificações bem sucedidas dentro de 30 s de funcionamento, com o código QR “O” posicionado de forma que só ele aparecesse nas imagens da câmera, sem necessidade de calcular tamanho, distância ou posição, num nível de intensidade luminosa de 1275 *candelas* (cd) de uma sala de estudo, apropriadamente convertidas de seu valor em lux, conforme ABNT NBR ISO/CIE 89950-1 (2013).

Com base na situação controle, foi analisado como o dispositivo se comporta em três aspectos: posição do código QR nas imagens da câmera, altura do dispositivo em relação ao piso e como a intensidade luminosa afeta a leitura dos códigos QR. O primeiro conjunto de testes visou identificar o código QR em seis áreas, chamadas aqui de “setores”, dentro do campo de visão da câmera, posicionando o código QR dentro de cada setor, com o dispositivo imóvel. O segundo grupo de testes manteve o código QR imóvel no piso enquanto foi ajustada a altura do dispositivo em relação ao chão em intervalos de 10 *centímetros* (cm), variando a altura de 1,0 a 1,4 *metros* (m) do chão. O terceiro e último conjunto de testes se utilizou de lâmpada *spot* halógena com intensidade luminosa máxima de 1000 cd conectada a um *dimmer* para verificação da capacidade de identificação em sete níveis de 58 a 1000 cd. Ambos dispositivo e lâmpada foram fixados em tripés sobre uma mesa e posicionados em direção ao piso, com sistema a 1,2 m e a lâmpada a 2,35 *centímetros* do chão. Cada alteração de variável foi testada 50 vezes. Mensurações que apresentaram erros por ruído foram retiradas da análise.

Na etapa seguinte foram realizados os testes móveis, feitos em um corredor de 22,85 m, onde foram criados dois trechos: o primeiro de 13 m e o segundo de 11 m. Foram posicionados obstáculos próprios do ambiente e códigos QR referentes a caminhos a seguir, obstáculos a contornar e fim do caminho. No primeiro trecho foram testados fatores como atrasos e melhores configurações de captação de imagem e padrões vibratórios, de modo a testar o alcance e a

precisão da detecção executada pelo sistema. No segundo trecho, foi testada a eficácia do sistema em diferentes velocidades de andar. A primeira parte do teste foi executada com auxílio de monitor para verificar as imagens adquiridas pela câmera e ajustes necessários no tempo das respostas, enquanto a segunda parte foi realizada com e sem monitor. Para simular o movimento a longo do corredor, o sistema foi colocado sobre uma mesa móvel e analisou-se as respostas à medida que se aproximava de códigos QR posicionados ao longo do piso do corredor.

Na última etapa, foi desenvolvido o colete que abriga todos os componentes de forma portátil. Depois, o sistema foi colocado no colete confeccionado e testado em ambiente laboratorial. O teste foi executado com o intuito de verificar o funcionamento do sistema, a leitura das informações pelo usuário e se é confortável para uso.

2.5 ANÁLISE ESTATÍSTICA

Para compreensão de como eram distribuídos dos dados adquiridos durante os testes, foi utilizado o teste de normalidade Shapiro-Wilk e foi encontrado que a distribuição dos dados se deu de maneira não-Gaussiana. Portanto, os dados adquiridos foram representados pelos valores de mediana ou intervalo interquartil (IIQ). O nível de significância estatística foi equivalente a $P < 0,05$.

3 CAPÍTULO 1: BASES TEÓRICAS

3.1 MOTIVAÇÕES

Este trabalho é um aprofundamento dos estudos feitos sobre os efeitos positivos que a tecnologia pode prover à arquitetura e urbanismo, através de novas formas de interação entre o usuário e o ambiente, principalmente no que se refere ao uso de objetos capazes de se conectarem entre si através da internet ou capazes de reconhecer padrões específicos já programados, para assim permitir um cotidiano mais cômodo aos residentes e transeuntes de determinado espaço.

O papel da arquitetura se estabelece como uma criação humana para humanos: a criação e adequação de espaços naturais para servir a funções e práticas humanas, a utilidade deve ser primária a qualquer obra arquitetônica, como é possível ler em PULS (2006), ao ponto que um edifício construído que não cumpre sua função está fadado ao desuso e consequente abandono.

Portanto, numa realidade onde a tecnologia se faz mais presente no dia-a-dia, a integração desta no ambiente construído também será consequente e certa. Do mesmo modo, a sabedoria para se utilizar destas novas ferramentas se torna uma consequente obrigação, a fim de que se tenha controle sobre elas, não se deixar controlar por esses objetos.

Isso foi bastante abordado no Trabalho de Conclusão de Curso do autor deste trabalho, que visou a criação de um sistema de alertas conectado a pulseiras com sensores de queda para ser utilizado em uma instituição de longa permanência para idosos (ABDON, 2019). O intuito desse trabalho foi desenvolver um sistema capaz de auxiliar os profissionais da saúde a chegar mais rapidamente a um indivíduo caído e integrá-lo ao próprio edifício. Portanto, pode ser visto que duas partes são essenciais ao projeto, para funcione: o objeto a ser utilizado pelo indivíduo, que tem o direito de aceitar utilizar ou não o objeto, e o espaço de convívio onde o objeto complementar do sistema deve ser instalado, onde, em virtude da natureza do ambiente, deve haver cuidados especiais para sua instalação.

Essa relação intrínseca entre humano e arquitetura é natural na sociedade moderna, onde estipulamos limites do que pode ser feito em dito local baseado em nossas próprias expectativas. O espaço construído se torna uma extensão do indivíduo a todo momento, portanto qualquer alteração, destruição ou adição deve ser comunicada e explicada àqueles que frequentarão, constantemente ou espaçadamente, tais locais. Ademais, quaisquer funções adicionais transformam o modo como as pessoas interagem com as construções, como mudanças sutis, como cores, iluminação e texturas, que são capazes de modificar como são vistos e vividos determinados ambientes, a modificações mais radicais, como paredes, pisos, rampas, escadas, tetos e janelas, mudam totalmente como indivíduos se comportam dentro destes locais. A introdução da tecnologia, mais notadamente a internet, também introduziu novas formas de interação: pessoas buscando sinais de *wi-fi* por quaisquer motivos tendem a mover-se em direção ao ponto onde o sinal é mais forte, aonde permanecem por maior período de tempo até completarem a tarefa em questão. O mesmo pode acontecer quando o sinal de internet é ruim dentro de determinado edifício, levando os indivíduos a saírem das construções em busca de um melhor sinal para prosseguirem com suas atividades.

Essa é apenas uma nova forma de interação entre humanos e arquitetura introduzida pela tecnologia dentre várias outras. Essa questão não somente foi abordada como explorada como base do trabalho prévio a este, uma vez que buscou-se desenvolver um objeto que adicionava uma nova interação entre indivíduo e construção: a de que o aparato arquitetônico pode servir para auxiliar profissionais a prestar socorro com maior eficiência, assim como uma

ação, ainda que inconsciente ou acidental, pode causar uma resposta para o ambiente onde o sujeito que a causou.

Portanto, a exploração das formas como os espaços construídos podem ser utilizados para interagirem com seus usuários estimulou a procura de novas questões possíveis de aliar à tecnologia assistiva, a fim de buscar alternativas para proporcionar conforto e tranquilidade aos transeuntes e residentes, e maior foco para como se utilizar da tecnologia assistiva para melhorar a convivência do ser humano com a arquitetura.

3.2 ARQUITETURA

Uma vez levantada no trabalho anterior, agora a questão da acessibilidade na arquitetura fora direcionada para os indivíduos com deficiências visuais, especialmente aqueles com cegueira total. A definição de cegueira pela Organização Mundial de Saúde se refere às pessoas com acuidade visual menor do que 3/60, campo de visão menor que 10° ou incapacidade de percepção de luz, conforme escrito na Classificação Estatística Internacional de Doenças e Problemas Relacionados à Saúde ou CID 11, também chamada de *International Classification of Diseases* ou *ICD 11* em inglês (*World Health Organization*, 2019). Em um estudo realizado em 2010, a Organização Mundial de Saúde estimou que havia 285 milhões de pessoas com deficiências visuais, das quais 39 milhões eram consideradas portadoras de cegueira (*World Health Organization*, 2013).

Essa parcela da população possui uma forma de interação com o espaço construído extremamente delicada, ao ponto que é necessária a adição de elementos táteis que os orientem para e pelas construções onde frequentam. Desde modo, a arquitetura em si detém de um papel muito importante para o cotidiano destes indivíduos: os de lhes prover de informações suficientes que lhes permitam autonomia e segurança de locomoção e direção a todo momento.

O exemplo mais aplicado da acessibilidade na arquitetura e urbanismo para deficientes visuais é o uso dos pisos táteis nos passeios, passarelas e demais ambientes de transição, conforme regulamentados no item 5.4 da ABNT NBR 9050 (2015) e na ABNT NBR 16537 (2016). A ABNT NBR 9050 também apresenta modelos diversos de sinalização tátil a serem aplicados nos objetos urbanos e arquitetônicos, como uso de sinalização de portas e passagens, mapas táteis ou sonoros para orientação de locais, rotas e outros elementos geográficos, placas de sinalização de pavimentos com números em braile usadas em conjunto com sinalização em braile nos corrimãos, assim como a sinalização dos degraus de uma escada, rampa ou

elevadores. Porém, mesmo com a legislação vigente, não é possível perceber o cumprimento destas normas com o devido rigor no espaço público brasileiro, especialmente no âmbito da cidade de Belém do Pará, onde é realizado este estudo.

Além destes elementos, se considerarmos objetos tecnológicos típicos de uma *smart home*, os quais podem se conectar entre si e com os seus usuários, a definição de acessibilidade com auxílio da alta tecnologia possibilita diversos modelos flexíveis para os mais diferentes tipos de necessidades. Uma definição para *smart home* seria a aplicação de alta tecnologia em residências para monitoramento e eficiência de energia, gerenciamento de energias renováveis, sistemas de saúde, serviços avançados de multimídia e vigilância, conforme conceituação de BATALLA, VASILAKOS e GAJEWSKI (2017), que ressaltam a interação dos habitantes dessas residências com os objetos de maneira diferenciada, uma vez que estes objetos não mais são artigos tecnológicos isolados, mas uma rede de comunicações que acomodam as preferências individuais dos usuários, montando ecossistemas próprios de cada lar.

MARIKYAN, PAPAGIANNIDIS e ALAMANOS (2019) ainda ressalta que tais *smart homes* são modelos residenciais que provém benefícios à saúde, através de instrumentos que permitem maior autonomia e conforto a pessoas idosas ou com necessidades específicas, e também benefícios ambientais e financeiros, pois através dos artigos tecnológicos mais eficientes e do monitoramento do uso de energia, permite-se maior controle dos diversos equipamentos e serviços instalados, de modo a reduzir o consumo energético residencial. Adicionalmente, apresenta-se neste estudo problemas comuns neste tipo de residência, como a questão da barreira tecnológica, que se refere à facilidade de uso dos sistemas autômatos e da confiabilidade e consistência do seu funcionamento. Outro problema apresentado são as preocupações financeiras, éticas e legais, uma vez que esses sistemas possuem alto custo de entrada, mas, em virtude da constante aquisição e troca de informações entre os objetos e entre a internet, podem ferir de diversas maneiras a privacidade do indivíduo, assim como as questões legais se tornam incertezas e são trabalhadas em maioria caso a caso, uma vez que são artigos muito recentes e de funcionamento distinto entre si para que seja feito rapidamente uma padronização. Por fim, a questão do próprio conhecimento de uso desses artigos tecnológicos são uma barreira inata, assim como a confiança das pessoas nestes sistemas, que, se mal utilizados, podem gerar dependência e isolamento por parte de seus usuários (seja por preferência pelo ambiente virtual ou pela segregação econômica), assim como as pessoas podem se recusar a se adaptar a estes novos sistemas por força de hábitos preexistentes.

No campo da acessibilidade, as tecnologias das *smart homes* são utilizadas das mais diversas maneiras. Um exemplo é citado no trabalho de SOUSA e MAIA (2014), que aborda

variadas adaptações possíveis e aplicadas para auxiliar residentes idosos portadores da doença de Alzheimer: um deles se apresenta como modelo de iluminação noturna em LEDs ao redor de portas para auxiliar orientação pelos ambientes durante o período noturno e sistema de controle de entrada de luz através de cortinas e lâmpadas (FIGUEIRO, 2008, *in* SOUSA e MAIA, 2014). Outro modelo estudado foi o uso de realidade aumentada e *Internet of Things* (“internet das coisas”) para auxiliar pessoas idosas residentes de *smart homes* no trabalho de BELEN, BEDNARZ e FAVERO (2019), onde foi desenvolvido um sistema baseado em dispositivo HoloLens para indicar a localização de itens pessoais, como óculos e chaves, e móveis, como cadeiras. O usuário seria então alertado via aviso auditivo quanto à identificação do item e veria o sistema adicionar uma marcação dentro do feedback visual. O usuário poderia então pedir por orientações para chegar ao objeto e o sistema montaria um trajeto, considerando possíveis obstáculos pelo caminho.

O uso da tecnologia para solucionar problemas observados nas mais diversas esferas, particulares ou públicas, ainda é uma prática nova, principalmente se considerar a própria arquitetura, cujas construções e adaptações não são fáceis ou rapidamente aplicáveis. Para a entrada de um edifício, por exemplo, rampas de acesso devem ser consideradas desde o projeto inicial, e caso contrário, deverão ser planejadas e adequadas à construção finalizada, considerando-se atentamente as necessidades, vias de acesso e quaisquer fatores importantes que acomodem os indivíduos que farão uso constante dessas vias acessíveis. O mesmo se aplica às tecnologias assistivas: se há o intuito de se construir um edifício que permita acesso indiscriminado a todos, sejam residentes ou visitantes, as tecnologias empregadas deverão ser consideradas durante o projeto inicial, no mais tardar serem estudadas com cautela e instaladas devidamente em casos de prédios já finalizados, como seria o caso em reformas ou adaptações.

3.3 TECNOLOGIA ASSISTIVA

No texto legal *PUBLIC LAW 100-407-AUG. 19, 1988*, dos Estados Unidos, conceitua-se pela primeira vez que “tecnologia assistiva” é aquilo que permite pessoas com deficiências a terem maior autonomia sobre suas vidas, participar e contribuir plenamente em sua vida privada e pública, melhorar interação com outros e se beneficiarem de quaisquer oportunidades que quiserem. A renovação desta lei, *PUBLIC LAW 105-394-NOV. 13, 1998*, somente reforçou a importância destes sistemas na vida de todas as pessoas em todas as esferas, inclusive foi identificado que o progresso feito nessas tecnologias facilita a vida de todos e que devem ser

reforçados e continuados quaisquer programas que permitam estes objetos a continuarem a melhorar a vida dos cidadãos. Por outro lado, no Brasil o próprio termo “tecnologia assistiva” é muito novo na esfera legal, sendo abordado pelo Comitê de Ajudas Técnicas (CAT), constituído pela portaria nº 142 de novembro de 2006, que caracteriza o termo como área de conhecimento interdisciplinar que engloba um grupo de produtos e serviços que visem a autonomia e participação de pessoas com deficiências, incapazes ou com mobilidade reduzida (BRASIL - SDHPR. – Comitê de Ajudas Técnicas – ATA VII, 2007, *in* BERSCH, 2017).

Embasado nesses conceitos e nas necessidades observadas, principalmente no que se refere à locomoção e orientação, a tecnologia assistiva aplicada aos sistemas das *smart homes* deve se focar no desenvolvimento de sistemas que incorporem sensores que reconheçam textos escritos sem braile, obstáculos, desníveis, caminhos, ambientes e sinalização, e consigam repassar essas informações através de alertas auditivos ou táteis preferencialmente.

Estudou-se o estado da arte no decorrer de toda a duração deste trabalho, em busca das melhores alternativas e modelos que estavam em desenvolvimento, onde foram analisados sistemas de auxílio a pessoas com deficiência visual de diversas maneiras: sistemas vestíveis, que dirigem a pessoa ativamente, que apenas orientam seus usuários, que apenas os alertam sobre seus arredores, sistemas que se utilizam de diferentes tipos de tecnologia, *hardware* ou *software*, entre outros. Os estudos levados em consideração foram aqueles apresentados em análises de estado da arte feitos por SANTOSO e REDMOND (2015), TAPU *et al* (2018), PAIVA e GUPTA (2019), ZAFAR *et al* (2022) e MADAKE *et al* (2023), nos quais foi verificado um número crescente de trabalhos desenvolvidos utilizando-se algoritmos que trabalham com visão através de computador, sistemas portáteis auxiliados por sensores diversos ou modelos de inteligência artificial. Com base nas considerações gerais feitas nesses estudos, foi decidido o modo como deveria ser desenvolvido o projeto deste trabalho: seria desenvolvido um algoritmo fechado que pudesse funcionar autonomamente, num sistema portátil que se orientasse o usuário através de câmeras para detectar caminhos e obstáculos. Acima de tudo, o sistema desenvolvido deveria ser de fácil manuseio, manutenção e baixo custo.

Deste modo, para se desenvolver um produto voltado para guiar indivíduos com deficiência visual, tipo cegueira total, foram levantados diversos trabalhos a fim de buscar a melhor forma de trabalhar a tecnologia a desenvolvida. Um destes foi o trabalho de TJAN *et al* (2005), que desenvolveu um sistema de sinais digitais próprio para guiar deficientes visuais, sinais estes lidos por uma câmera especialmente adaptada. Este sistema optou por este modelo de guia através de detecção de imagens de modo a não fazer com o que o objeto escolha o caminho para o indivíduo, mas que permita a ele ter noção dos seus arredores e escolher por

si só o caminho que deseja seguir. O trabalho de TJAN, portanto, influenciou bastante o modo como foi decidido desenvolver o sistema deste trabalho: escolheu-se permitir ao usuário do objeto desenvolvido a fazer suas próprias decisões com base nas informações adquiridas pelo processamento virtual e repassadas a ele por meio de algum mecanismo de respostas.

Trabalhos que se utilizam de *smartphones* também foram analisados a fim de esclarecer o modo como é feita a identificação de obstáculos e compreensão de layouts de ambientes e como foram ajustados os aparelhos junto ao indivíduo. O estudo de JEON *et al* (2012) buscou uma forma de auxiliar indivíduos cegos a compreender o layout de um cômodo não familiar através do uso de uma câmera vestível e aplicativo de celular, onde o usuário acionaria o modo de escaneamento do ambiente, feito pela câmera, e obteria respostas acerca de objetos considerados importantes, como mesas, cadeiras e outros móveis e objetos de escritório, assim como outros celulares. O estudo de TAPU *et al* (2013) consistiu no desenvolvimento de um algoritmo que analisa imagens de vídeo em tempo real para detecção de obstáculos, estáticos ou dinâmicos, lentos ou rápidos, utilizando-se um smartphone como câmera e sistema de processamento, sendo este preso num suporte de peitoral a ser utilizado pelo indivíduo. Esse trabalho obteve sucesso ao analisar ambientes variados, sem necessidade da criação de sinais específicos para alertar o indivíduo, utilizando-se da relação de movimento e tempo entre os diferentes de pontos de referência das imagens adquiridas. Semelhantemente, o trabalho de CHACCOUR e BADR (2015) também se utiliza de celulares modernos para contribuir para este tema, mas de outra forma: o aparelho está conectado à uma rede de comunicação que se utiliza de câmeras instaladas no ambiente interno que capturam as imagens e as enviam para serem processadas por um computador, que então as analisa e envia respostas apropriadas para o *smartphone* particular do usuário, sem necessidade de portar qualquer aparelho se não seu próprio celular particular ou aprender alguma informação que se refira somente ao sistema.

O modo como os sistemas emitem respostas para o indivíduo foi bastante estudado a fim de achar o modelo menos invasivo ou irritante, assim como o que melhor possui capacidade de emitir sinais específicos e claros. Foram considerados modelos de respostas em áudio, sejam palavras faladas ou sinais sonoros, e sistemas de vibração, por estes emitirem as respostas somente ao usuário de maneira discreta e não intrusiva. Os trabalhos de WACHAJA *et al* (2015, 2017) foram de grande importância para o entendimento do mecanismo de respostas vibratórias, ressaltando de diferentes maneiras como motores vibratórios transmitiam informações aos usuários do sistema inteligente desenvolvido: um andador comum adaptado com notebook para processamento das informações. Os motores alertavam o usuário quanto aos mais diversos tipos de obstáculos, fossem eles negativos como escadas para baixo, ou

suspensos como fitas, além de guiar os indivíduos alertando sobre a direção para onde deveriam dobrar para evitar obstáculos estáticos ou dinâmicos. As vibrações eram emitidas somente nas alças do andador no trabalho de 2015, mas foram adicionados motores vibratórios a um cinto aumentar a precisão das respostas no trabalho de 2017. Um modelo semelhante, mais robusto e especialmente construído para esta tarefa, pode ser visto em JIMÉNEZ (2018), no qual fora construído um andador inteligente com alças adaptadas para sustentação do usuário, portanto aumentando a superfície de contato e aumentando a possibilidade de percepção das vibrações. O trabalho de LEE e IN (2023) se aprofunda em respostas vibratórias ao desenvolver um sistema de pulseiras com um arranjo complexo e tridimensional de motores vibratórios para a criação de respostas específicas para diversos tipos de mensagens que o sistema deve passar ao usuário, permitindo-os grande alta precisão em orientar os usuários em testes práticos em uma rota de mais de 700 m só para pedestres. As respostas vibratórias também foram estudadas no sistema de PATIL *et al* (2018), onde foi desenvolvido um modelo baseado em sensores ultrassônicos para serem usados no nível do piso montados em botas, de modo a detectar através destes sensores a presença de obstáculos à frente e emitir respostas vibratórias a motores nas próprias botas. O sistema foi bem-sucedido na detecção de diversos tipos de obstáculos na altura do piso até a altura do joelho, inclusive escadas ascendentes, porém foi ineficaz em detectar objetos negativos, tais como escadas ou rampas descendentes e buracos, a menos que se pusesse sensores nas solas. O sistema também era capaz de distinguir superfícies secas de molhadas, porém somente quando o usuário pisava em poças, limitando a possibilidade de evitar acidentes, mas ainda capaz de alertar o usuário sobre a superfície molhada. Em MADAKE *et al* (2023), foi desenvolvido um sistema de percepção de arredores com rede neural baseado na plataforma Nvidia Jetson, com sensores ultrassônicos e câmera acoplados para detecção de obstáculos, e motores vibratórios e fone de ouvido singular para emissão de respostas, com todo o sistema acoplado a um colete. O sistema teve sucesso em emissão de respostas e descrições precisas aos usuários, permitindo-os ter noção do ambiente em que se encontram e como podem se orientar para chegar ao seu destino.

Algo importante a se ressaltar, porém, foi a decisão de se utilizar de produtos mais acessíveis para o desenvolvimento de um sistema com custo baixo. Além da questão da portabilidade, considerou-se outras questões como importantes: o produto a ser desenvolvido deve ter potência para analisar o ambiente e emitir respostas em tempo real, deve ser capaz de processar as informações sozinho para minimizar possíveis problemas de comunicação e custos de manutenção, assim como deve ser um sistema que permita um tempo de vida útil longo, tanto em termos de *hardware* quanto de *software*. Dentro destes parâmetros, diversas

plataformas possíveis, baseadas principalmente em microcontroladores e microcomputadores, como visto no estudo de TEPELEA *et al* (2019) e no de KLEINBERG *et al* (2023), cujos trabalhos são baseados na plataforma Raspberry Pi.

O estudo de TEPELEA *et al* utilizou-se de sinais de trânsito como imagens-base dentro do sistema, os quais foram cadastrados para serem reconhecidos, correlacionados e comparados com imagens captadas em tempo real por uma câmera associada à placa Raspberry Pi 3 Model B+ e algoritmo construído com base em bibliotecas OpenCV. O modelo de respostas escolhido foi através de sinais acústicos via fone de ouvido. O artigo levou em consideração três resoluções utilizadas, tempo de processamento pelo sistema e distância de detecção dos sinais, tanto em ambientes fechados quanto em ambientes abertos. Os resultados se demonstraram bem-sucedidos em reconhecer os sinais de trânsito a distâncias aceitáveis e demonstrou que a plataforma e as bibliotecas utilizadas foram capazes de lidar com a tarefa em todos os ambientes testados.

O estudo de KLEINBERG *et al* (2023), por outro lado, utiliza Raspberry Pi 4 B em conjunto com sensores ultrassônicos e câmera com aprendizagem de máquina para detectar obstáculos, e emite respostas através de óculos conectados por Bluetooth e motores vibratórios. O dispositivo criado funciona em tempo real e se utiliza de um colete aonde os componentes são posicionados junto ao corpo do usuário. O artigo fez testes de detecção de obstáculos com usuários e obteve resultados promissores quanto a prover a maior percepção dos arredores e permiti-los navegar independentemente em segurança.

3.4 COMPONENTES E CUSTOS

Com base na análise dos trabalhos pesquisados e da problemática a ser trabalhada, foi considerada a plataforma onde seria desenvolvido o sistema pretendido. Com base nas exigências listadas anteriormente, viu-se a necessidade de um microcomputador com sistema operacional próprio, portátil, modular e suficientemente capaz de processar imagens em tempo real. A plataforma escolhida para tal projeto é a de microcomputadores da marca Raspberry, semelhante ao utilizado em TEPELEA *et al* (2019).

Se utilizada como um computador comum, esta plataforma fornece um objeto capaz de rodar tarefas cotidianas como organizar arquivos, escrever e ler textos, navegar pela internet e fazer projetos simples. Ademais, por se utilizar de sistema operacional e *softwares* abertos ao público, isto é, aberto ao público, o sistema é muito utilizado para variados projetos científicos

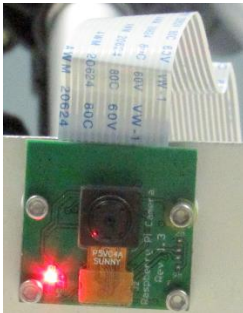
e experimentais, como sistemas automatizados e robóticos. Ele também possui interface modular própria e de fácil acesso para permitir expansão de suas funções, como adição de peças como motores, caixas de som, sensores e câmeras.

Estabelecida a plataforma, partiu-se então para consideração dos modelos disponíveis para uso neste projeto. Utilizou-se a placa Raspberry Pi 3 Model B+, que já estava disponível em mercado brasileiro no início do desenvolvimento da pesquisa e era fornecida oficialmente em território nacional desde 2018 pela Agência Nacional de Telecomunicações (ANATEL). O preço desta placa quando adquirida foi de R\$ 280,00 *reais* (BRL), porém, devido à demanda por objetos tecnológicos e escassa produção de componentes causada pela pandemia, o custo a ser considerado neste trabalho é o preço de mercado em 2023.

Os itens escolhidos para o sistema desenvolvido, porém, podem ser substituídos por outros mais potentes e caros, ou menos potentes e mais baratos, como exemplo a própria placa Raspberry, que poderia ser substituída por sua sucessora: a Raspberry Pi 4, lançada em 2019. Ela possui processador mais rápido e moderno, configurações maiores de RAM e maior número de entradas de USB. Entretanto, ela está no mercado nacional a preços mais elevados que a placa utilizada neste trabalho, variando entre R\$ 675,00 a R\$ 1.000,00, a depender da configuração de RAM. Outra alternativa ainda dentro da plataforma Raspberry se encontra na placa Raspberry Pi Zero 2 W, que é significativamente menor, possui configuração menor de RAM e menos entradas de comunicação, mais ainda mantém os 40 pinos GPIO. Ela é mais barata que a Raspberry Pi 3, custando por volta de metade do valor, mas preferiu-se a configuração maior de RAM para o desenvolvimento deste trabalho.

Ademais, acessórios considerados para o desenvolvimento da pesquisa foram analisados e comprados conforme planejamento criado: uma câmera, dois motores vibratórios, módulo de bateria externa e quaisquer materiais necessários para construção do suporte a ser utilizado. Abaixo, tabela de itens utilizados, com nome, descrição, preço atual (2023) e imagens ilustrativas. Os preços estimados informados não consideram fretes e estão em real (BRL), oferecidos no mercado brasileiro, e *dólar americano* (USD), que seria preço de importação. Produtos exclusivos do mercado nacional ou produtos gerais serão convertidos para dólar.

Nome	Descrição	Preço Estimado	Imagem Ilustrativa
------	-----------	----------------	--------------------

Cartão micro SD	Cartão de memória formato micros de 32 GB para instalação do sistema operacional Raspbian.	50,00 BRL 10,00 USD	-
Raspberry Pi 3 Model B+	Microcomputador da marca Raspberry, processador Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit de até 1.4 GHz e 1 GB DDR2, além de possuir entradas para vídeo em HDMI, 4 portas USB, micro SD, ethernet, Wi-Fi e 40 pinos GPIO.	500,00 BRL 35,00 USD	
Case para Raspberry Pi 3	Case protetora dissipadora de calor em alumínio preto para Raspberry Pi Model B+, com duas ventoinhas.	80,00 BRL 16,00 USD (convertido)	
Motor de Vibração 1027	Motor vibratório em moeda de diâmetro 10 mm, altura de 3 mm, máximo de 9000 RPM, Corrente máxima: 90 mA tensão de 2.5 V a 4 V. Duas unidades.	7,00 BRL un. 0,50 USD un.	
Módulo de Câmera da Raspberry V1	Módulo de câmera de 5 Megapixels, modos de vídeo de 1080p a 30 QPS e 720p a 60 QPS, foco fixo, profundidade de campo de aprox. 1 m ao infinito, distância focal de 3.60 mm +/- 0.01, campo de visão horizontal de 53.50° (+/- 0.13°) graus e vertical de 41.41° (+/- 0.11°) graus.	70,00 BRL 25,00 USD	
Bateria Externa VX CASE, 24549 – CB0301	Capacidade de 9600 mAh, saída de 5 V e corrente de até 4.5 A, entrada & saída USB-C, entrada USB, peso de 258 gramas.	500,00 BRL 100,00 USD (convertido)	

Material de Impressão 3D	Rolo de Filamento PLA.	100,00 BRL 20,00 USD (convertido)	-
Colete em alças	Alças de mochila de tecido nylon preto, esponja, engates.	150,00 BRL 30,00 USD (convertido)	
Total	-	1464,00 BRL 237,00 USD	-

Tabela 1. Lista de produtos utilizados no sistema desenvolvido. Imagens registradas pelo autor.

4 CAPÍTULO 2: DESENVOLVIMENTO DO ALGORITMO

4.1 PLANEJAMENTO INICIAL

Os desenhos abaixo ilustram o conceito do objeto. Na figura 2 é proposta a organização preliminar dos componentes no colete. Tal organização foi escolhida de modo que a placa Raspberry Pi e a câmera se encontrem em posição central em relação ao usuário, impedindo identificação tendenciosa para um dos lados.

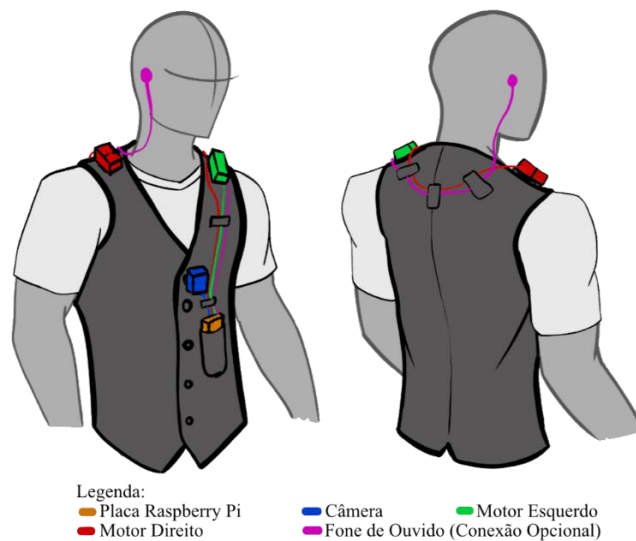


Figura 2. Esquema preliminar. Imagem criada pelo autor.

Ao se propor o local escolhido para os motores vibratórios, foram considerados posicionamentos nos braços ou na cintura. O primeiro caso foi descartado devido a extensão de cabos possibilitar a ocorrência de acidentes que possam danificá-los ou desconectá-los em choques mecânicos. A posição ao longo do ventre ou cintura também foi descartada devido à possibilidade que o próprio colete e a vestimenta do usuário atrapalhem na compreensão dos sinais vibratórios. Considerando esses possíveis problemas, foi escolhido o posicionamento nos ombros devido à relativa curta extensão dos cabos e também porque nesta posição deve haver pouca interferência na identificação dos avisos vibratórios. Para alcançar os dois ombros, os cabos inicialmente foram posicionados ao longo das costas do indivíduo, mas isso foi redesenhado no protótipo final do colete.

Também é indicado na imagem uma possível alternativa para os motores vibratórios nos ombros, na forma de um fone de ouvido conectado diretamente à placa Raspberry Pi, que emitiria as respostas como frases faladas. A instalação deste sistema alternativo não foi testada neste estudo, mas pode ser uma opção em futuros trabalhos.

Na figura 3 é demonstrada a posição e inclinação proposta para a câmera. A escolha da obtenção de imagens direcionada para o piso em frente ao usuário vem da preocupação com possíveis choques contra obstáculos mais próximos. Deste modo, para haver melhor detecção destes, a câmera do objeto será direcionada ao chão, abrangendo um espaço que esteja sempre dentro do tempo de caminhada do indivíduo. A inclinação exata e a área abrangida foram melhor experimentadas na fase de testes, assim como a altura a ser posicionada no colete.

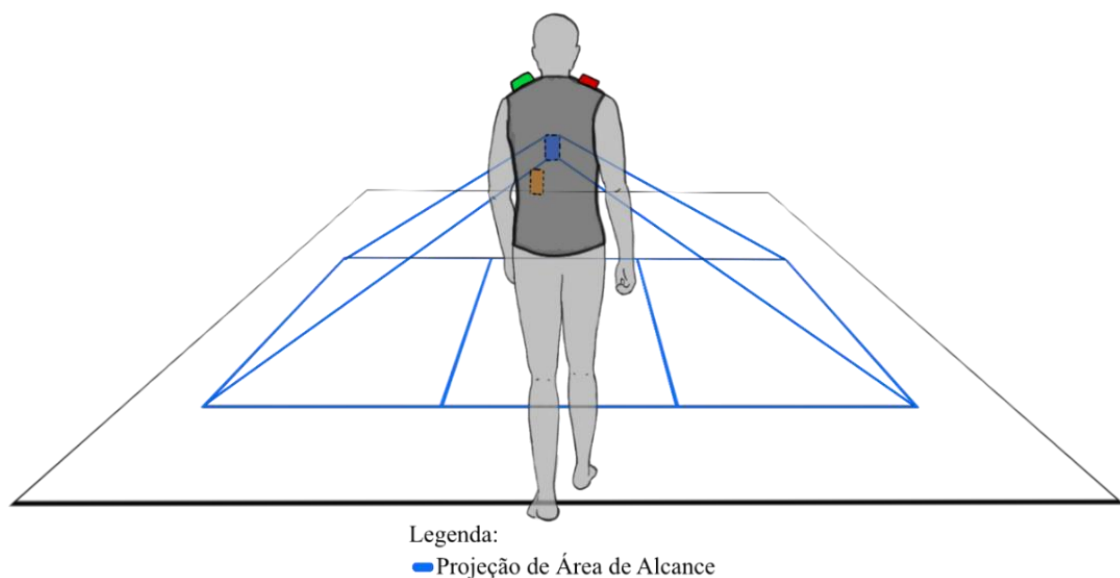


Figura 3. Planejamento do alcance da câmera. Imagem criada pelo autor.

Foi planejado que os sinais visuais sejam posicionados no chão (figura 4) e sigam o padrão fornecido pela ABNT para pisos táteis, a norma ABNT NBR 16537, podendo ser utilizado em conjunto com esse sistema em lugares aonde estão presentes. Também foi desenhado como o posicionamento dos sinais visuais do tipo “obstáculos” poderiam ser posicionados, descrito na figura 5: com as marcações no piso, ao redor do objeto indicado.

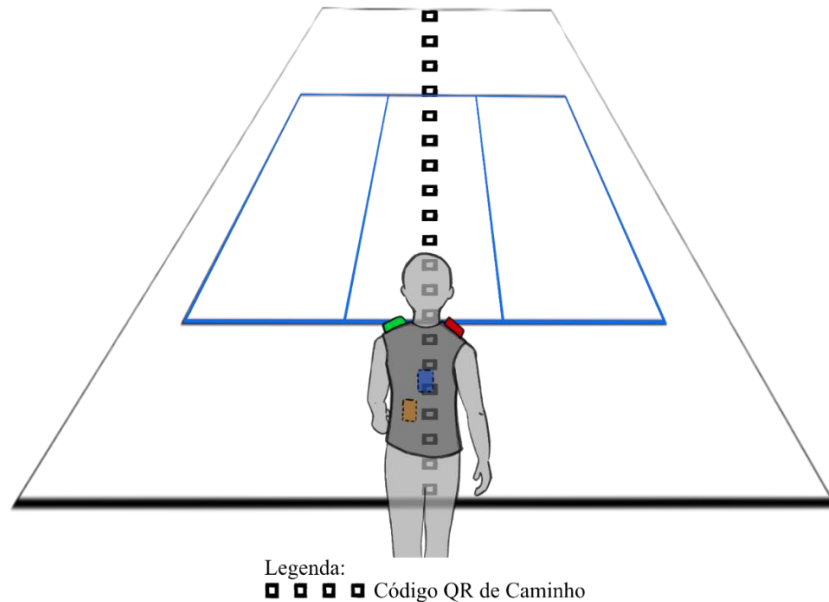


Figura 4. Planejamento da situação padrão. Imagem criada pelo autor.

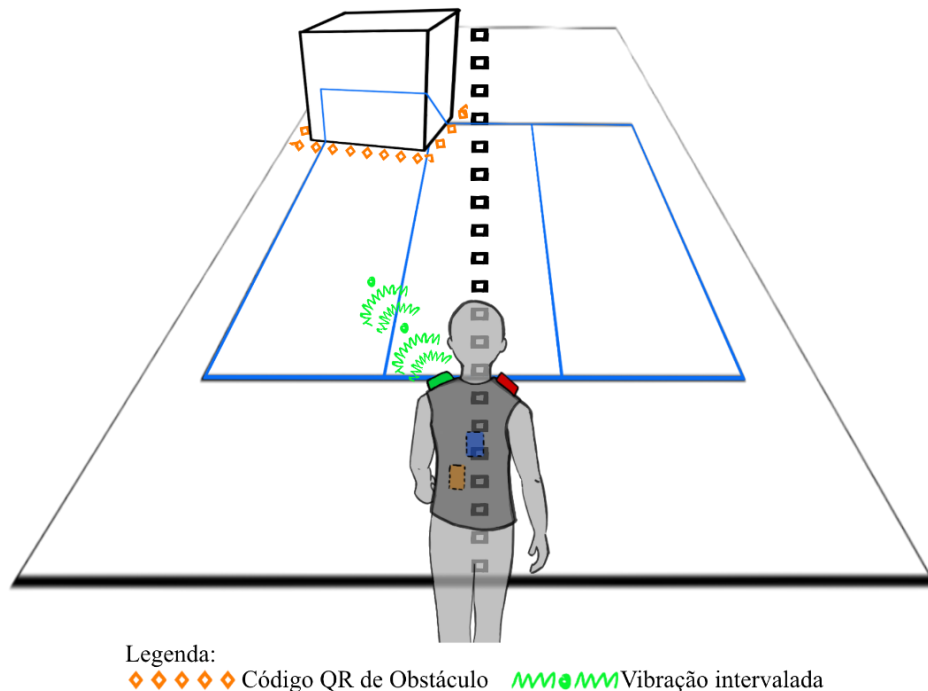


Figura 5. Planejamento de detecção de um obstáculo. Imagem criada pelo autor.

4.2 MONTAGEM PRELIMINAR DO SISTEMA

Após a formulação inicial do código, os componentes foram testados a fim de confirmar o funcionamento da câmera e dos sinais QR. Para tal, a placa Raspberry foi conectada a um monitor e foi alimentada por entrada USB 3.0 de um computador (figura 6-A), com a finalidade de interação direta com o sistema operacional. A câmera utilizada para os testes iniciais foi uma webcam da marca Logitech, modelo C270 (figura 6-B), conectada por entrada USB.

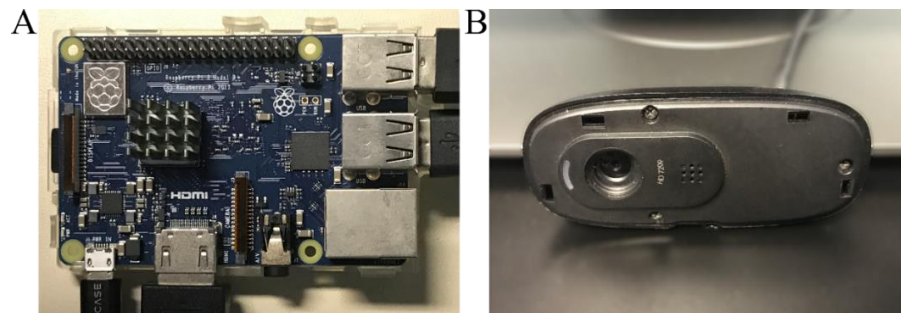


Figura 6. Raspberry Pi 3 Model B+ utilizada (A). Webcam Logitech C270, fixada na mesa (B). Imagens registradas pelo autor.

Para os códigos QR, fora utilizado o site *The QR Code Generator* (disponível em: <<https://www.the-qrcode-generator.com/pt/>>) para criar os sinais de forma gratuita. Os textos referentes a cada sinal foram inicialmente traduzidos para inglês simplificar a emissão de respostas, uma vez que foram identificados erros quando o código programado precisava representar caracteres com acentos. Os códigos QR foram impressos em papel A4 em tamanho 12,7 cm x 12,7 cm para os testes de identificação, distância efetiva do sistema, definição de prioridades de respostas. Os sinais criados foram referentes a: “*obstáculo*”, “*caminho à frente*”, “*caminho à esquerda*”, “*caminho à direita*”, “*fim do caminho*”. Foi impresso mais um sinal em QR para representar a condição “*end*” no código, a fim de finalizar os testes (figura 7).

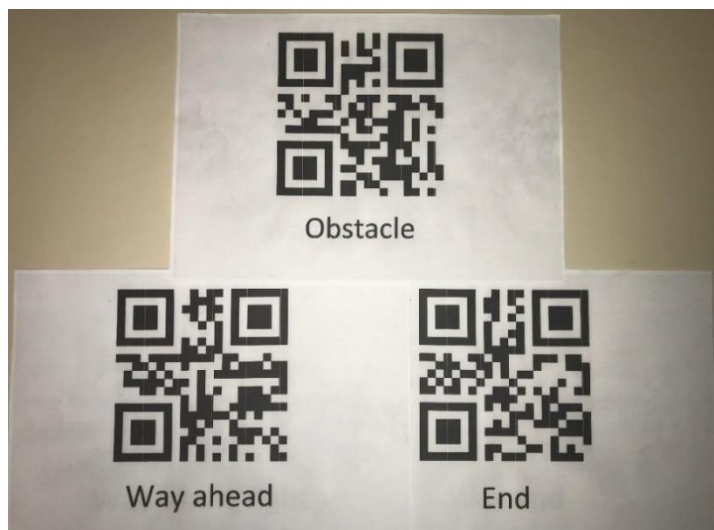


Figura 7. Sinais visuais em código QR Obstacle (“obstáculo”), Way ahead (“caminho à frente”) e End (“fim”). Imagem registrada pelo autor.

4.3 CONSTRUÇÃO DO ALGORITMO

O algoritmo (Apêndice A) pode ser dividido em três etapas: definição dos padrões de identificação de sinal QR, definição de parâmetros da câmera e emissão de respostas.

Para a leitura dos códigos QR, foi utilizado um modelo que pode ser encontrado no tutorial “*Building a Barcode/QR code Reader using Python*”, de GUVEN (2020), que utiliza uma câmera para identificar um sinal QR impresso. Por si só, esse trecho não gera resposta em vídeo, apenas usa a câmera para fazer a leitura das informações. Para tal, foi necessário um adendo que interaja diretamente com o dispositivo de captura de imagem e repasse o conteúdo captado através de uma janela de transmissão ao vivo. Neste feedback é possível ver que o trecho do código responsável pela identificação gera um quadrado ao redor do código QR identificado com sucesso, descrevendo o conteúdo em branco logo acima (figura 8-A). Foi verificado que o sinal QR também pode ser identificado caso rotacionado (figura 8-B, 8-C).

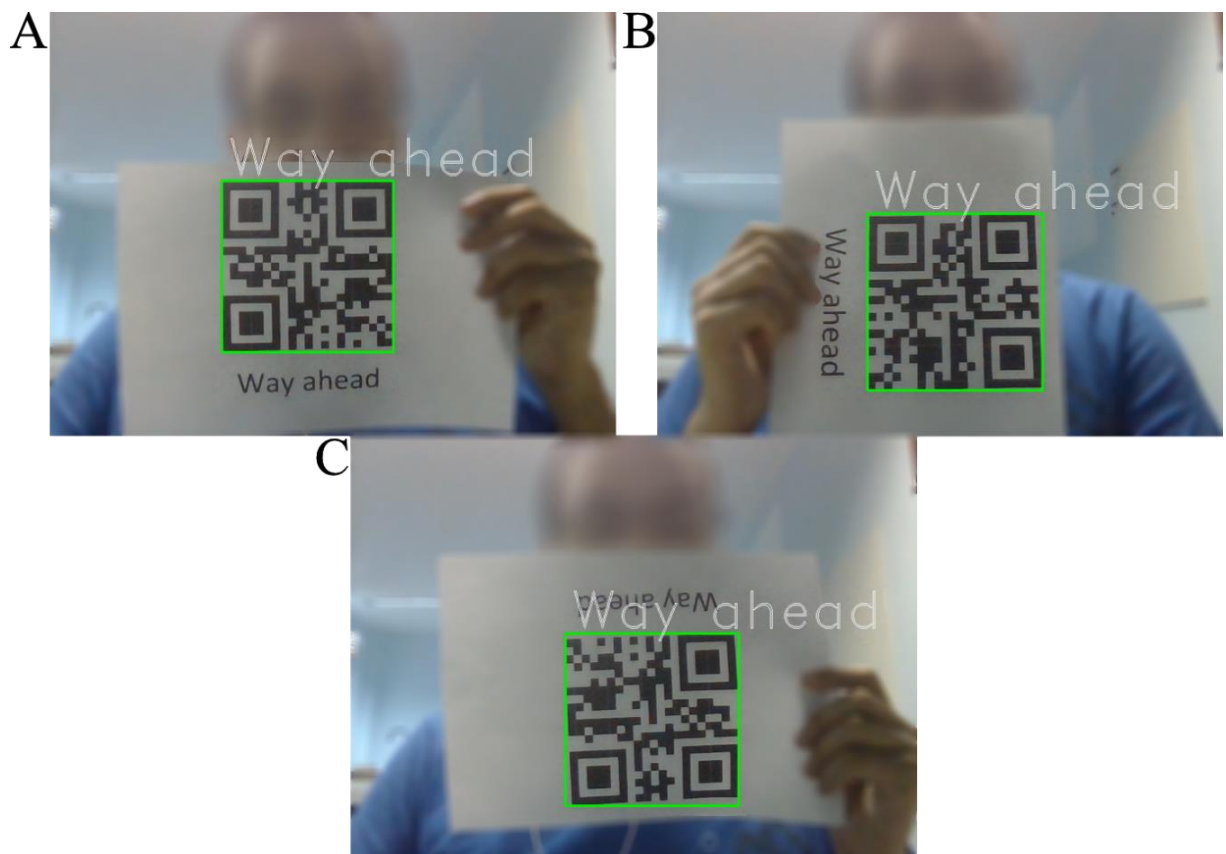


Figura 8. Exemplo de identificação bem-sucedida de código QR e feedback da câmera em janela (A) e de identificação bem-sucedida de sinal QR rotacionado (B, C). Imagens registradas pelo autor e editadas para seletivamente borrar o fundo de modo a enfatizar o funcionamento do dispositivo.

Entretanto, como este trecho do código só é capaz de identificar, ele não de armazena informações para gerar respostas futuras. Para o armazenamento das informações identificadas, foi necessário criar uma condição “*with*” responsável por gerar um arquivo chamado “*barcode_result.txt*”, aonde é armazenado em tempo real somente os códigos QR identificados. Esse trecho também apaga as informações do arquivo sem não houver identificação de sinal QR pela câmera, evitando o acúmulo de respostas a cada ciclo de leitura do código.

Como o código é visado para uso em placas Raspberry Pi, é de suma importância que as imagens não atrasem o processamento das informações dos sinais QR, de modo que não afetem as respostas em tempo real de modo significativo. Para tal, foi escrito no algoritmo linhas que possibilitam alterar a resolução e o filtro das imagens de modo a utilizar menos processamento. Abaixo, um exemplo em *640 pixels de largura × 480 pixels de altura* (480p 4:3) e vídeo em preto e branco, sem tons de cinza (figura 9), o que também eliminou de penumbras da imagem, gerando maior número de identificações bem-sucedidas em câmeras ligadas por USB, como webcams.



Figura 9. Feedback em resolução menor e em preto e branco gerado na webcam Logitech C270. Imagem registrada e editada pelo autor.

As respostas emitidas mediante aquisição das informações presentes nos dados dos sinais QR é a última etapa do código. Esta etapa está inserida como condição “*with*” do texto que lê as informações da câmera. Essa condição lê o conteúdo do arquivo “*barcode_result.txt*” gerado na etapa anterior e, através de uma condição “*if*”, checka se o conteúdo confere com algum dos comandos registrados. Se houver leitura de sinal QR que não esteja registrado no código, não haverá emissão de resposta.

As respostas registradas nos primeiros testes foram *Obstacle* (“obstáculo”), *Way Ahead* (“caminho à frente”), *Way to the left* (“caminho à esquerda”), *Way to the right* (“caminho à

direita”) e *Way end* (“*fim do caminho*”). Para testar a emissão de respostas antes de acoplar o sistema de motores vibratórios, foi escrito o comando “*print*” para registrar no terminal do programa frases representativas do tipo de vibração que seria utilizado para determinado sinal QR (figura 10). Para melhorar a eficácia dos testes iniciais, foi adicionada uma condição “*if*” que lê o sinal QR *End* (“*fim*”) para finalizar o loop do código e facilitar correções, porém essa condição foi removida na etapa de testes com os motores.

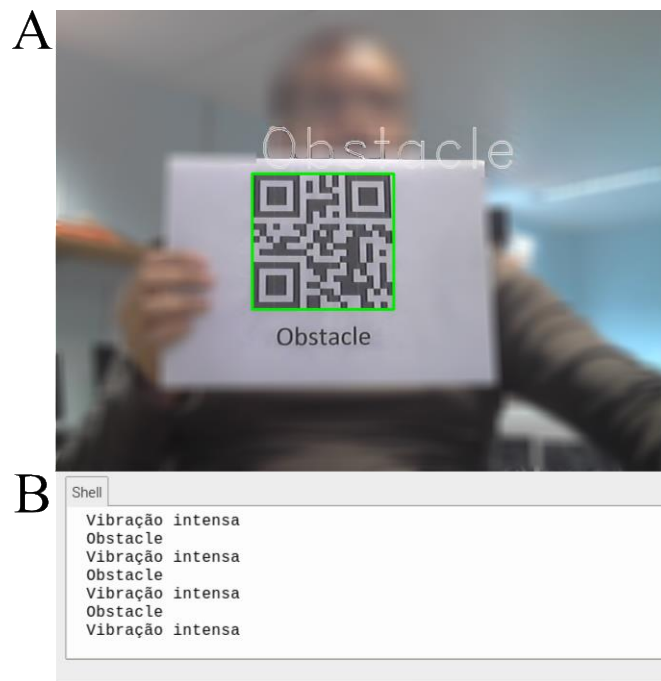


Figura 10. Exemplo de respostas emitidas perante identificação de código QR. Imagem registrada e editada pelo autor.

4.4 AJUSTES DE IMAGEM, PRECISÃO E DISTÂNCIA

Para melhor ajustar as imagens para detecção com suficiente precisão e distância para a correta emissão de respostas, foram realizados em ambiente laboratorial com o objeto conectado a um computador por USB para prover energia e a um monitor via HDMI para feedback visual e adaptações do algoritmo em tempo real. A câmera foi fixada sobre a tela e múltiplos sinais visuais em código QR foram impressos em papel A4. Esta configuração fora escolhida de modo a estipular como seria considerada a precisão do sistema e verificar a distância necessária para identificar com sucesso os códigos impressos, assim como simular situações específicas relacionadas à câmera, como iluminação do ambiente ou múltiplos códigos QR em uma determinada imagem.

Nesta etapa, a precisão do sistema estava ligada ao número de respostas ininterruptas escritas no terminal do software de programação. As respostas foram consideradas “precisas” quando não houve lacunas em branco entre as identificações bem-sucedidas dos sinais visuais, assim como foram consideradas respostas “não precisas” quando houve intervalos nas identificações registradas no terminal (figura 11).

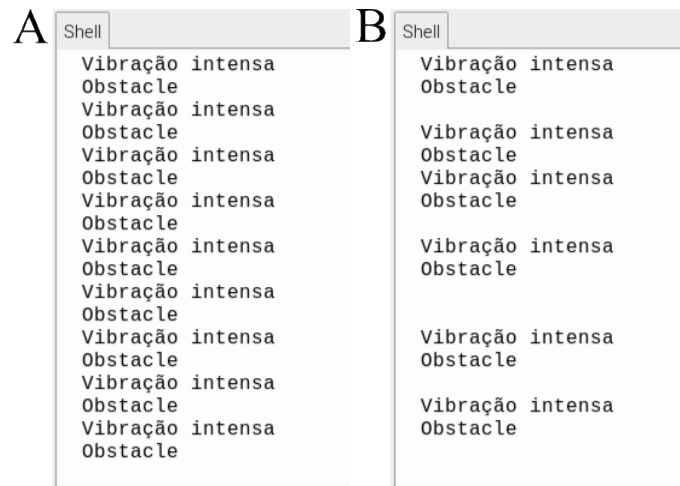


Figura 11. Respostas “precisa” (A) e “não precisa” (B). Imagem registrada pelo autor.

A distância entre a câmera e o sinal visual também foi um fator importante na precisão das respostas. Utilizando imagens coloridas em 480p 4:3 e dentro de uma sala de laboratório iluminada com luzes LED (aproximadamente 1275 cd de intensidade luminosa), a distância mínima para identificação bem-sucedida foi de aproximadamente 21 cm entre a câmera e o sinal visual, enquanto a distância máxima foi de cerca de 2,37 m (figura 12-A, 12-B e 12-C).

Esses valores sofrem modificações quando se altera certos parâmetros do sistema, como cores da imagem, resolução e câmera. Um exemplo se traduz na troca do filtro das imagens adquiridas pela câmera de colorido para preto e branco absolutos. Esse filtro funciona retirando o canal RGB e analisando somente o canal de níveis de cinza, mapeado de 0 a 255, onde 0 é completamente preto e 255 é completamente branco. Então são definidos os limiares, aonde são convertidos os pixels de 0 a 127 para totalmente preto, e de 128 a 255 para totalmente branco, eliminando assim o contraste gradual em tons de cinza e aplicando um filtro de contraste absoluto que resulta em uma imagem apenas em preto e branco. A aplicação deste filtro em câmeras conectadas por USB gerou melhorias na precisão, aumentando a distância máxima verificada em imagens coloridas de aproximadamente 2,37 m para de 2,6 m (figura 12-F).



Figura 12. Imagens coloridas (A, B, C) e em preto e branco (D, E, F) referentes às distâncias de 21 (A, D), 120 (B, E), 237 (C) e 260 (F) centímetros. Imagens registradas e editadas pelo autor.

O uso de resoluções de câmera maiores possibilita maior precisão nos resultados devido à maior quantidade de pixels representando a imagem, porém ao mesmo tempo necessita de mais processamento da placa, o que pode afetar o tempo de uso do dispositivo em modo portátil ou a quantidade de imagens por segundo, atrasando a emissão de respostas. Usando a webcam da Logitech, as resoluções *1280 pixels de largura × 960 pixels de altura* (720p 4:3) e *1920 pixels de largura × 1440 pixels de altura* (1080p 4:3) aumentaram a distância máxima das imagens coloridas para 2,7 m e 3 m respectivamente, porém demoravam entre 1 a 4 s a mais para completar o loop do algoritmo, assim afetando a leitura em tempo real.

Semelhantemente, trocar a câmera também apresentou efeitos nos resultados: utilizando-se a câmera do notebook do autor (LG Gram, modelo 2018), o alcance máximo das imagens em 480p 4:3 e 1080p 4:3 foram de 1,5 m e 1,8 m respectivamente.

A definição do loop do código também fora ajustada para equilibrar o processamento da Raspberry com a identificação em tempo real. Nesta etapa, intervalo entre loops foi definido em 0,05 s, proporcionando a leitura das imagens da câmera a 10 QPS durante emissão de

respostas, e foi limitado a 30 QPS quando não há emissão de respostas, para que o sistema seja mais rápido em emitir a primeira resposta necessária para o alerta. Este tempo foi baseado na velocidade das passadas apresentada em SCHIMPL *et al* (2011), onde foi verificada a velocidade média de 1,25 m/s (4.5 km/h) para indivíduos saudáveis de variadas idades, e justificado em CLARK-CARTER (1986), que fala que pessoas com deficiência visual em média andam em velocidades mais baixas que pessoas com visão normal. Portanto, quando há identificação do sinal visual entre 1 a 2 m de distância, a taxa de atualização do código em 0,1 segundo se configura como suficiente para identificação e emissão de respostas antes que o usuário alcance a fonte do aviso em questão.

Na situação em que múltiplos QRs apareçam ao mesmo tempo nas imagens, verificou-se que o sistema se comporta conforme o tamanho dos códigos QR: os maiores são priorizados em comparação aos menores. O mesmo vale para sinais visuais de tamanhos iguais, porém em distâncias diferentes, uma vez que o tamanho real não é o considerado, mas o tamanho aparente nas imagens da câmera. Desta maneira, para contornar possíveis erros de prioridade de identificação, foi decidido padronizar o tamanho real dos códigos QR, de modo que o mais próximo do sensor da câmera sempre tenha tamanho relativo maior (figura 13).



Figura 13. Situações onde os sinais “Way ahead” (A) e “Way to the left” (B) estão mais próximos da câmera em diferentes momentos. Imagens registradas e editadas pelo autor.

Nesta etapa, também se definiu o nível de interação com o usuário e segurança do sistema, onde foi estabelecido que o usuário só poderá interagir diretamente com o sistema se tiver acesso físico a ele. O algoritmo não precisa de internet para funcionar, portanto foi decidido desativar tal capacidade como forma de prevenir intervenções oriundas de comunicação online, como links maliciosos. O sistema funcionará com dados cadastrados no

próprio algoritmo, ou seja: só emitirá respostas que estiverem contidas em sua base de dados, sem adição de novos códigos em tempo real. Para alterar, adicionar ou remover respostas emitidas, será necessário adicionar novas linhas de código e um novo código QR que representem a frase desejada. Isso pode ser feito ao conectar o sistema a um monitor para visualização e reescrever as necessidades nos trechos adequados.

4.5 MOTORES VIBRATÓRIOS E DEFINIÇÃO DE POSIÇÃO E DISTÂNCIA

Uma vez instalados na placa Raspberry dois motores vibratórios em moeda tipo 1027, seu funcionamento foi configurado conforme os resultados adquiridos da leitura de códigos QR específicos, apresentando emissões padronizadas para cada resposta. Essas emissões foram divididas por posição do código QR nas imagens adquiridas e por seu tamanho aparente. Neste sentido, a posição define que motor vibra, e o tamanho aparente define a intensidade do padrão vibratório. O motor 1 (motor esquerdo) foi conectado nas portas 33 (GPIO 13, PWM1) e 39 (terra), e o motor 2 (motor direito) foi conectado nas portas 32 (GPIO 12, PWM0) e 30 (terra), conforme pode ser visto no diagrama da figura 14, retirado do site de documentação oficial da Raspberry Pi 3 Modelo B+.

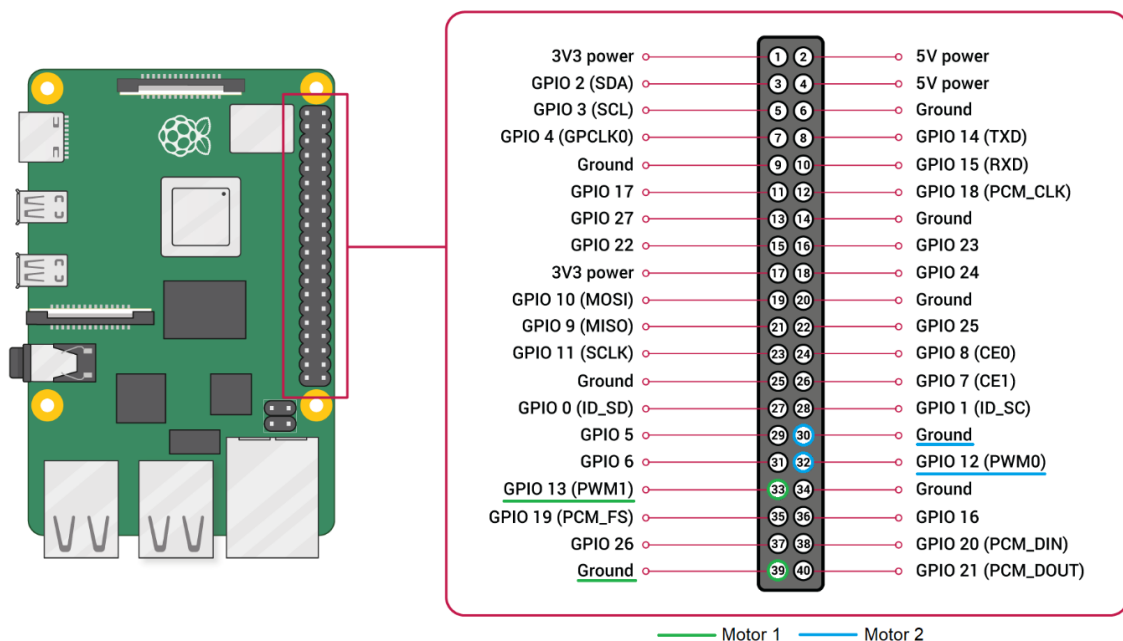


Figura 14. Diagrama das portas GPIO das placas Raspberry. Adaptado de *Raspberry Pi Documentation* (link para acesso: <<https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>>).

Para a definição da ativação dos motores, primeiro foi necessário definir como seriam divididas as posições do código QR dentro do algoritmo. Foi escolhido incorporar ao algoritmo uma condição que define áreas da imagem recebida pela câmera em setores, medidos em “pixels horizontais x pixels verticais”. Essa condição deve ser alterada de acordo com a resolução: para 480p 4:3, na qual o sistema foi inicialmente testado, foram criados três setores de 213 pixels de largura x 160 pixels de altura, e para *480 pixels de largura × 360 pixels de altura* (360p 4:3), uma segunda resolução a ser testada, os três setores de identificação foram reajustados para 160 pixels de largura x 120 pixels de altura. Os três setores foram posicionados lado a lado nas imagens, definindo área da esquerda, área do centro e área da direita. Os setores da esquerda e da direita correspondem aos motores a serem posicionados no lado esquerdo e direito do corpo do usuário, enquanto o setor do centro ativa respostas nos dois motores simultaneamente.

A divisão destes setores somente é visível ao sistema, não sendo mostrada na janela de feedback da câmera. Para a localização do código QR, é levado em consideração seu ponto central, por exemplo: na figura 15, o código QR está em dois setores diferentes (esquerdo e centro), mas seu ponto central está localizado no setor do meio, portanto o algoritmo reconhecerá que o código QR se encontra à frente, no centro. Portanto, a resposta emitida seria executada pelos dois motores ao mesmo tempo.

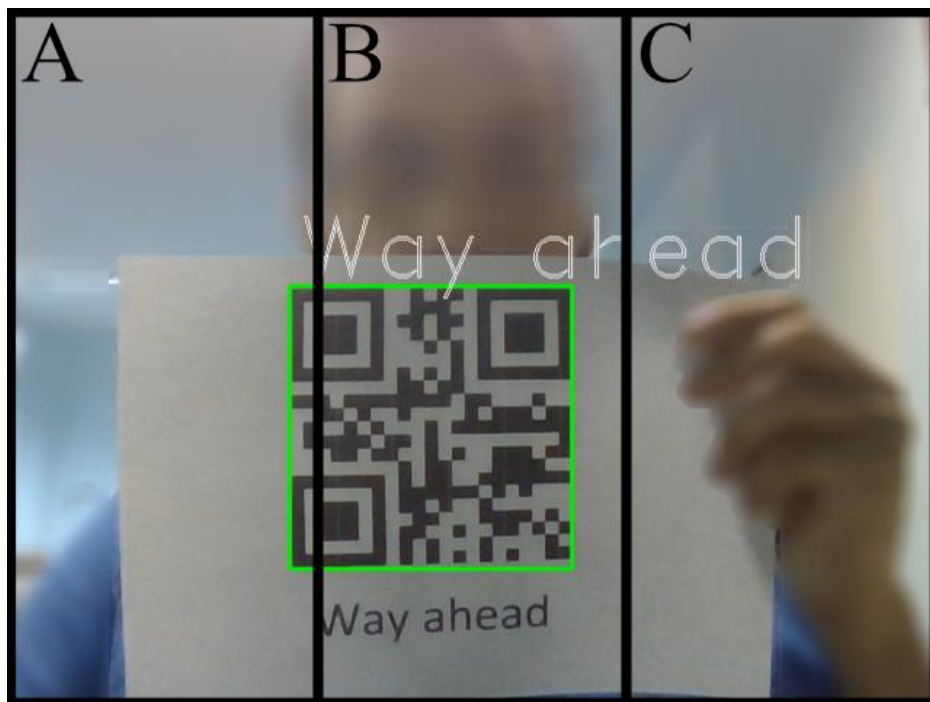


Figura 15. Representação visual da divisão dos setores. Imagem registrada e editada pelo autor.

Para a definição dos padrões vibratórios, modificou-se a parte do algoritmo referente à identificação de distância do item 4.4. O cálculo da distância valor é encontrado utilizando-se o tamanho aparente do objeto desejado e determinando o tamanho real e escala definida dentro do algoritmo, conforme é descrito em DAL (2021). Deste modo, e com a adição dos motores, pôde-se definir dois intervalos: a distância de até 1,5 m do código QR para a câmera é considerada “Perto” pelo sistema, enquanto distâncias acima disso são consideradas “Longe”. Tal esquema foi preferível em relação a três intervalos para eliminar a possibilidade de confusão entre diferentes níveis de potência das vibrações, importando ao sistema apenas alertar ao usuário quanto à existência de objetos à frente através de vibrações fracas na condição “Longe”, e alertar ao usuário a possibilidade real de choque contra objetos próximos através de vibrações intensas na condição “Perto”.

Somado às potências da vibração, utilizou-se a função “*sleep*” para criação de intervalos de 0,05 s, 0,15 s e 0,25 s entre cada ativação dos motores. Esses intervalos atrasam a conclusão do loop do código e prologam a vibração, criando assim vibrações mais curtas ou mais longas. A definição desse intervalo foi importante porque não poderia ser tão longa a ponto de interferir na identificação em tempo real, ou tão curta a ponto de passar despercebida ou gerar confusão no usuário. A combinação dos intervalos e das potências de vibração definem os diferentes tipos de emissão de respostas perante a identificação dos códigos QR pela câmera.

Com a criação dos setores, o número de códigos QR iniciais diminuíram: foram descartados os códigos “*à esquerda*” e “*à direita*” de cada tipo. Ao final desta etapa, decidiu-se manter apenas os tipos “*Obstáculo*”, “*Caminho*” e “*Final do Caminho*”, representados por códigos QR das letras “O” (Obstáculo), “C” (Caminho) e “F” (Final do Caminho), minimizando o número de informações lidas pelo sistema e acelerando o loop, tornando a leitura em tempo real mais rápida.

Portanto, cada um desses códigos é representado por intervalos específicos: a vibração intervalada de 0,15 s indica “*Obstáculo*”, a de 0,05 s indica “*Caminho*” e a de 0,25 s indica “*Final do Caminho*”. A potência das vibrações dos códigos QR “*Obstáculo*” e “*Caminho*” foi de 50% da velocidade dos motores (4500 RPM) para a condição “Longe” e 100% (9000 RPM) para a condição “Perto”, enquanto que para o código QR “*Final do Caminho*” foi utilizada 25% (2250 RPM) da velocidade do motor para “Longe” e 50% (4500 RPM) para “Perto”.

5 CAPÍTULO 3: MONTAGEM DO PROTÓTIPO E TESTES FINAIS

5.1 FUNCIONAMENTO DA CÂMERA EM POSIÇÃO FIXA

Antes de prosseguir para os testes finais, foi necessário estabelecer dentro do sistema operacional a forma como o objeto iniciaria o algoritmo ao ser ligado, uma vez que não haverá monitor para interagir no protótipo final.

O método utilizado aqui foi através da modificação de arquivos do sistema: o terminal abre automaticamente após entrar no sistema operacional através da alteração o conteúdo do arquivo “*autostart*”, encontrado em “*/.config/lxsession/LXDE-pi*”, e adição “*@lxterminal*” (in *StackOverflow*, 2016). Com o terminal abrindo automaticamente ao ligar o sistema, foi alterado o conteúdo do arquivo “*.bashrc*”, encontrado em “*/home/[nome do sistema]/.bashrc*”, e adicionadas as linhas “*echo Running at boot*” e “*python [caminho do arquivo a ser aberto]*” ao final, que dá ao sistema a capacidade de automaticamente ler um arquivo específico toda vez que for aberto o terminal (in *DEXTER Industries*). Dessa forma, ao ligar, o sistema operacional inicia o terminal, que então abre o arquivo do algoritmo criado neste trabalho (figura 16). Foi decidido também manter as frases por extenso escritas a cada resposta, de modo a melhor identificar possíveis erros se for necessário resgatar as informações quando houver interação direta com o sistema.

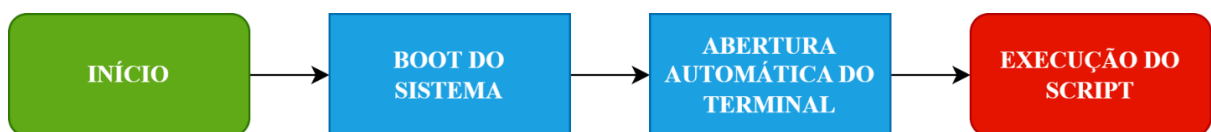


Figura 16. Fluxograma da inicialização do algoritmo desenvolvido. Imagem criada pelo autor.

Com algoritmo finalizado, foi necessário realizar cálculos a respeito da posição e inclinação da câmera do sistema a fim de obter os resultados adequados para a funcionalidade do objeto como desejado. Para tal, nesta etapa foram realizados testes em uma posição fixa para que fossem encontradas as distâncias de identificação dos códigos QR, já conectado em uma bateria portátil com saída de 5 V e corrente de 2.5 A para permitir portabilidade ao sistema.

Inicialmente, fora utilizada uma bateria de lítio RPi PowerBank V 1.0, que possui capacidade de 3800 mAh, saída de 5 V e corrente de 1.8 A, em virtude de ser recomendada para o uso em placas Raspberry Pi 3 Modelo B. Porém, apesar da capacidade não comprometer o funcionamento do sistema (apenas o tempo de uso), baterias com correntes menores que 2.5 A

afetaram a velocidade de processamento neste projeto: o sistema operacional e o algoritmo criado funcionaram, mas apresentaram *delays* significativos mesmo em resoluções baixas e filtro de imagem em preto e branco, assim comprometendo o funcionamento em tempo real.

No lugar desta foi utilizada uma bateria portátil de 259 *gramas* (g), com capacidade de 9600mAh, saída de 5 V e corrente de até 4.5 A, modelo 24549 – CB0301 da marca VX CASE. Com uso desta, o sistema foi capaz de se manter ligado por 6 a 7 horas de uso contínuo e funcionamento normal, o que condiz com a estimativa do consumo, uma vez que o uso total pelo projeto é de 1430 mA ($9600 \text{ mAh} / 1430 \text{ mA} = 6 \text{ horas e } 43 \text{ minutos}$). Com a corrente em valores adequados, o sistema pôde fazer as leituras em tempo real com *delays* aceitáveis nas resoluções escolhidas e não houve diferença significativa entre o consumo de bateria ou atraso de respostas entre os filtros de imagem.

Foi posicionado sobre uma mesa um tripé de alumínio convencional, no qual o sistema foi fixado ao tubo central para alcançar alturas de 1,0 a 1,4 m (figura 17). Foi colocada sobre a placa Raspberry um case de alumínio para proteção e dissipação de calor e então foi parafusada numa chapa de alumínio feita sob medida, onde também fora aparafusada a nova câmera a ser utilizada (figura 18). Os códigos QR impressos tiveram tamanho ajustado para 19,6 cm x 19,6 cm, para maior definição da imagem a ser analisada.

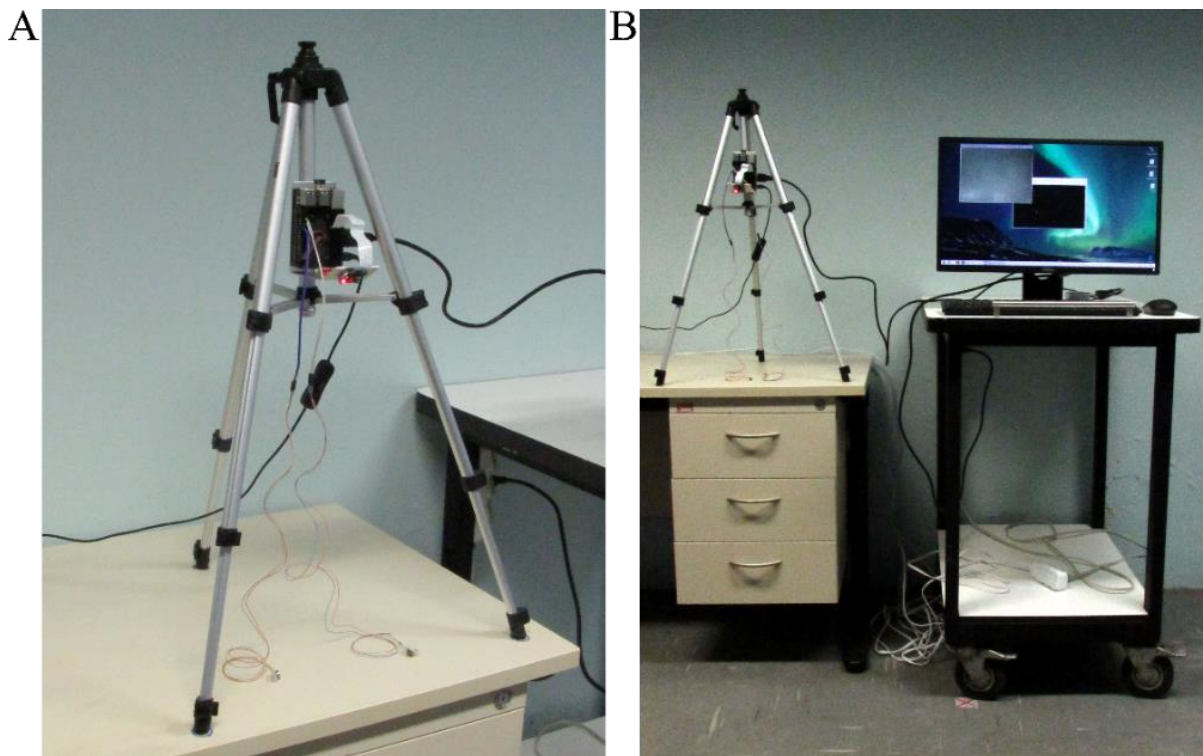


Figura 17. Imagens do sistema fixo, com a placa Raspberry Pi presa a um tripé (A), ligada a um monitor e alimentada por uma tomada (B). Imagens registradas pelo autor.

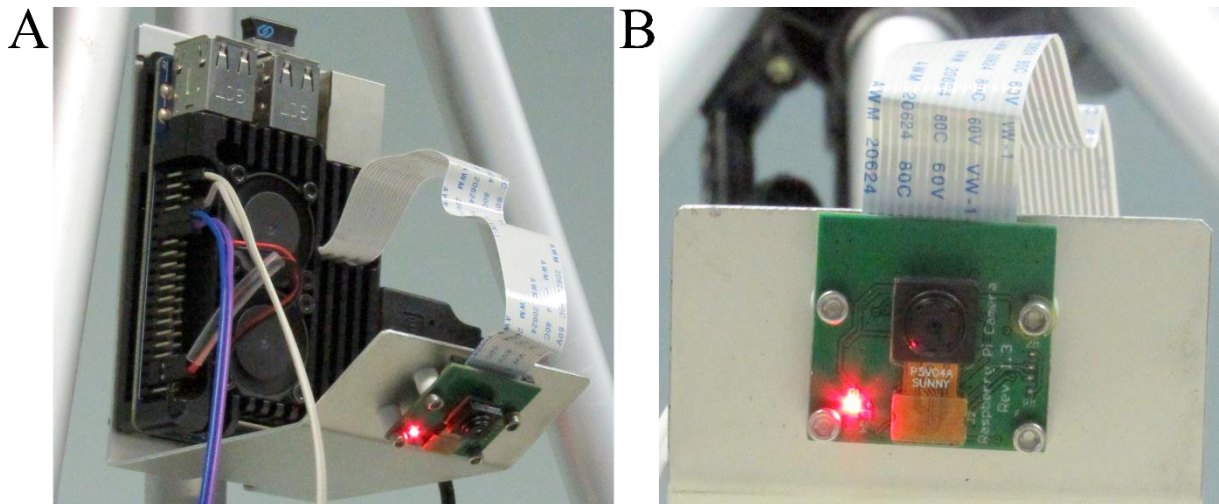


Figura 18. Detalhe da placa Raspberry Pi com case de proteção (A) e câmera fixada e rotacionada 180° (B).

Imagens registradas pelo autor.

A câmera USB foi trocada pelo “*Camera Module v1*” da Raspberry, chamado, por ser mais compacta, poder conectada diretamente à placa por uma entrada própria (eliminando o atraso natural da conexão por USB), ter resolução suficiente para manter objetos nítidos dentro do alcance do projeto e modos de vídeo com alta taxa de atualização de QPS. As especificações da lente da câmera também foram consideradas adequadas para o trabalho em questão. A câmera foi rotacionada em 180° para simplificar a montagem e diminuir a extensão dos cabos, então foi necessário rotacionar as imagens para o sentido correto dentro do software.

Para testar a precisão desta câmera e do dispositivo como tal, foi realizado um teste controle para ser comparado com variáveis que mais significativamente afetam o funcionamento do dispositivo, como altura do usuário, em que parte do campo de visão da câmera está o sinal visual e o nível de intensidade luminosa do ambiente. Para adquirir um valor base para comparação com essas variáveis, escolheu-se usar a resolução mínima em que o sistema consegue trabalhar, que é a resolução 360p 4:3, e posicionou-se o código QR “O” próximo ao dispositivo, a 21 cm da câmera, de modo que somente o código QR ocupasse o campo de visão, assim possibilitando maior nitidez do código a ser analisado. O ambiente (uma sala de estudo) possuía valor de intensidade luminosa de 1275 cd. Dentro dessa situação, foram feitos 50 testes para se achar o valor da mediana de identificações bem sucedidas dentro de um período de 30 s. O *controle* (CT), que apresentou mediana de 118 identificações a cada 30 s, foi considerado o valor de precisão base a ser comparado com os demais testes de altura, setor e nível de intensidade luminosa.

A câmera foi projetada para ser posicionada no peitoral do usuário e direcionada para o piso à sua frente, portando foi considerada inicialmente a distância de 1,2 m do chão, baseada

na altura média brasileira para homens e mulheres, 1,73 m e 1,6 m respectivamente (*NCD Risk Factor Collaboration (NCD-RisC), 2016*). Também foram analisadas alturas de 1,0 m, 1,1 m, 1,3 m e 1,4 m do chão, a fim de simular o posicionamento do dispositivo em indivíduos mais baixos e mais altos que a média apresentada. Devido à limites da própria área do campo de visão da câmera, da resolução de vídeo escolhida e das distorções devido à angulação do código QR em relação à câmera, os alcances máximos para identificação do código QR em 1,0 m e 1,1 m foram de 1,23 m e 1,3 m respectivamente, menores que a distância máxima encontrada nas alturas de 1,2 m a 1,4 m do chão, que apresentaram alcance máximo de 1,4 m (figura 19). Por outro lado, a taxa de identificação e emissão de respostas se apresentou estável e de acordo com a situação controle em todas as alturas, demonstrando que o sistema é capaz de identificar e alertar o indivíduo quanto aos sinais visuais em tempo e distâncias apropriadas (figura 20). Para os demais testes foi utilizada a altura de 1,2 m, tanto por ser o posicionamento desejado conforme a média brasileira de altura quanto por ser capaz de identificação na maior distância.

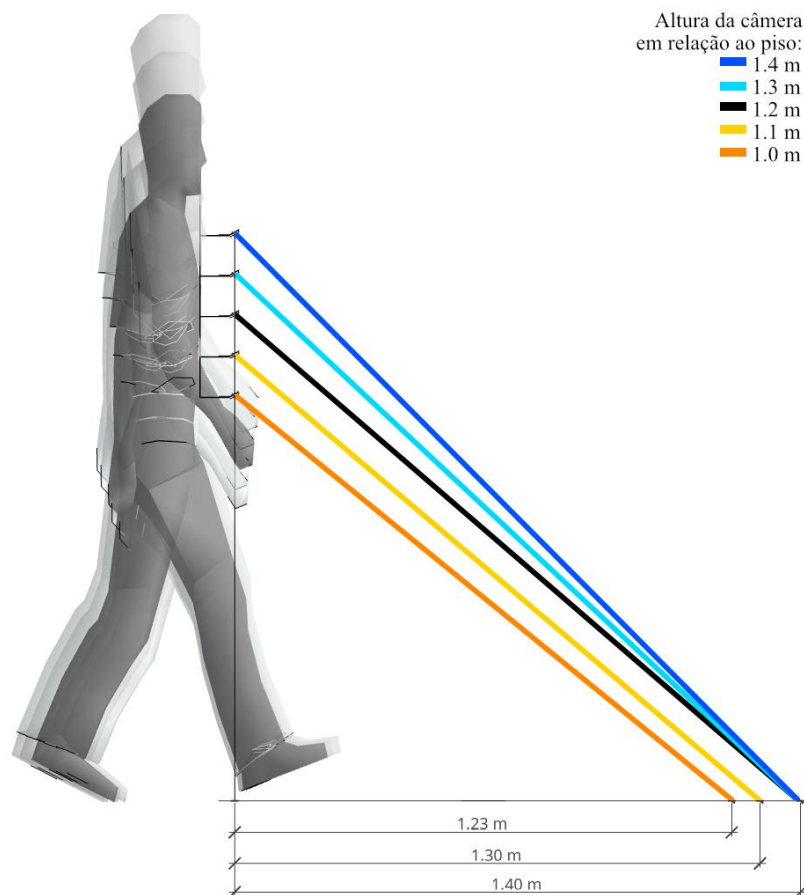


Figura 19. Distância máxima de identificação por altura verificada. Imagem criada pelo autor.

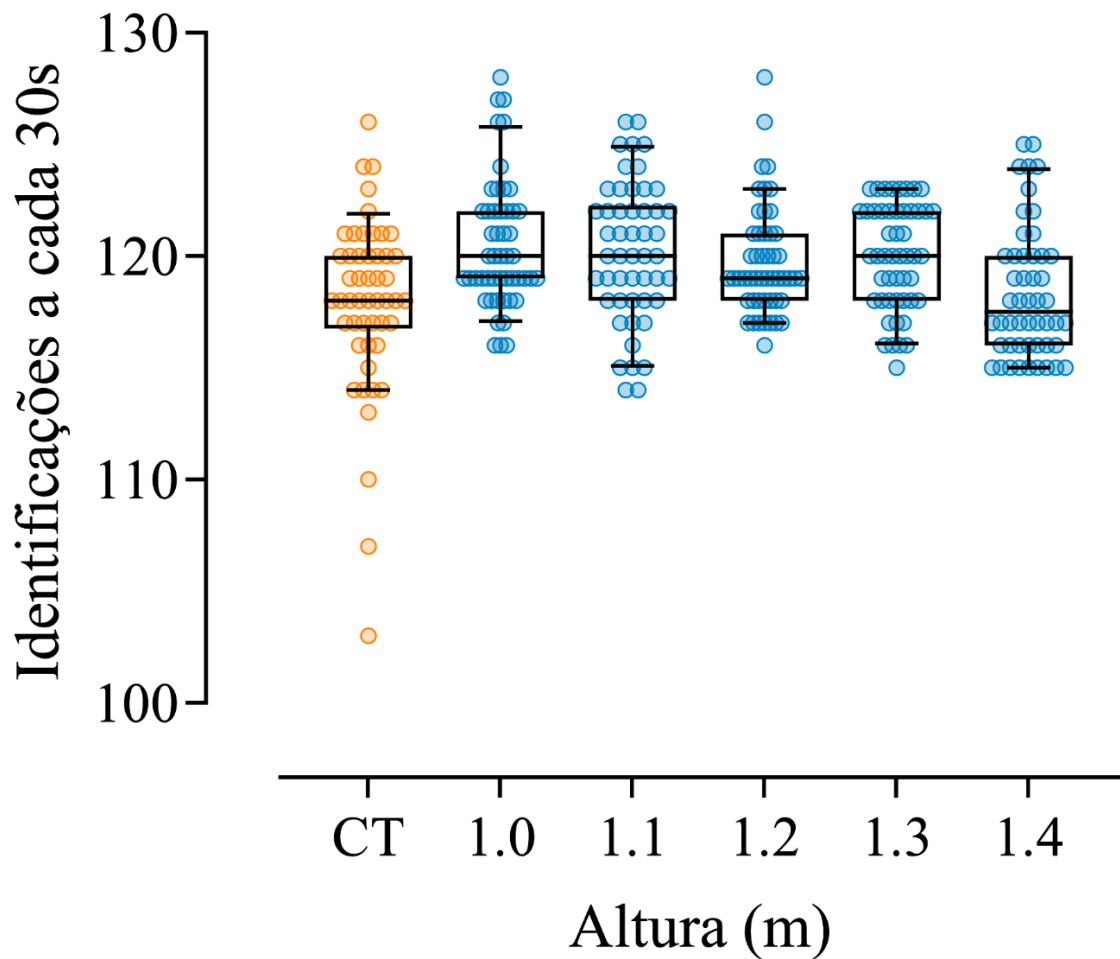


Figura 20. Taxa de identificação por altura da câmera, comparados ao controle (CT). Imagem criada pelo autor.

A angulação escolhida para as imagens da câmera foi de 30° em relação ao piso, levando em consideração que o sistema também deveria levar em conta objetos mais à frente do indivíduo, não devendo ficar totalmente paralelo ao chão. Com base na documentação da câmera usada (*in Camera, Raspberry Pi Documentation*), a área total possível de ser analisada é de $2,3 \text{ m}^2$ (área trapezoidal de 1,22 m na base menor, 2,02 m na base maior, 1,47 m em cada lateral e 1,42 m na distância entre as bases), a uma distância perpendicular de 23 cm à frente da câmera (Figura 22).

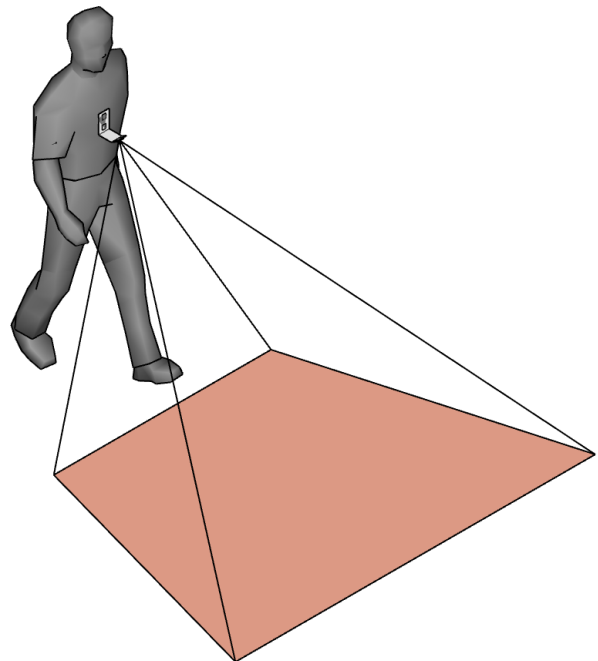


Figura 21. Simulação tridimensional do campo de visão da câmera. Imagem criada pelo autor.

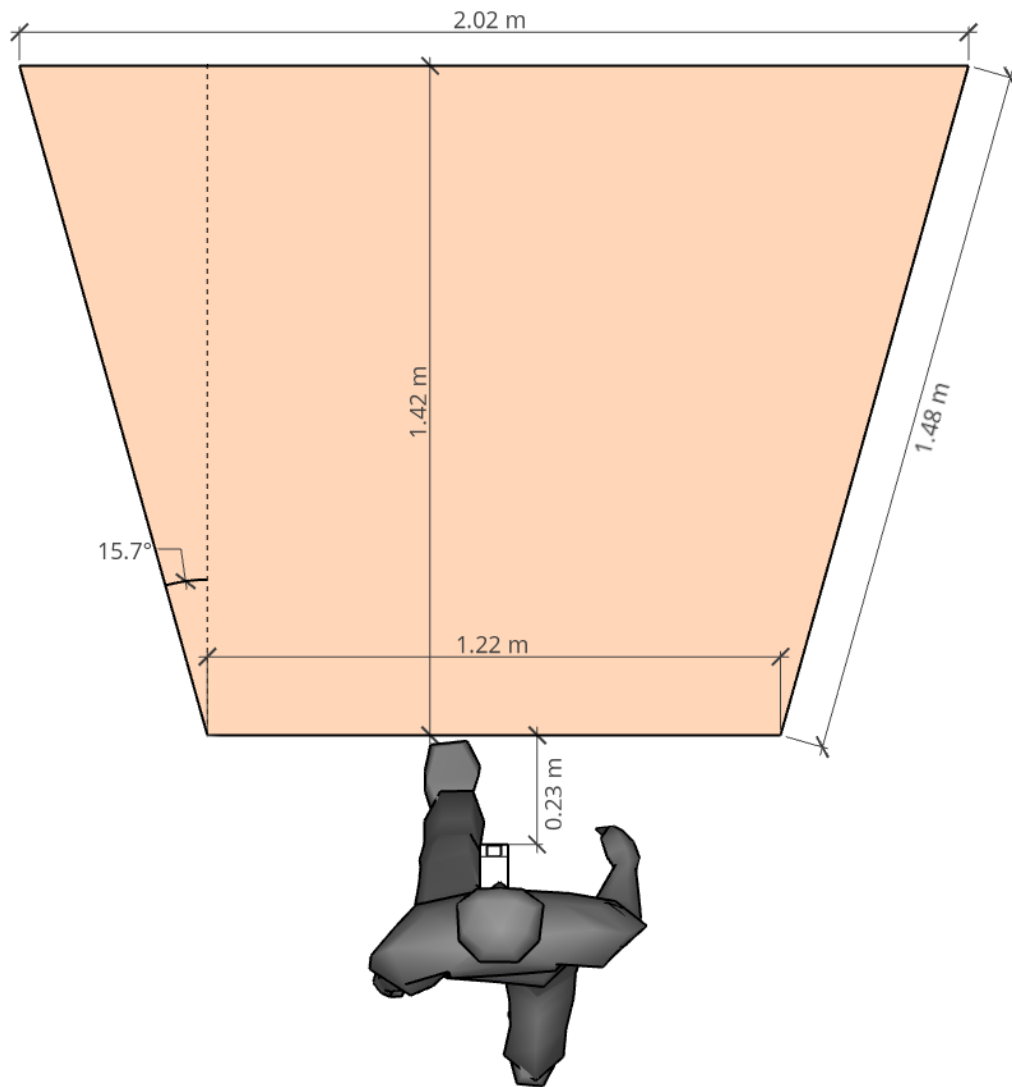


Figura 22. Projeção da área do campo de visão baseada nas especificações da câmera. Imagem criada pelo autor.

Outro fator importante a ser testado foi a precisão da identificação dos códigos QR em variadas distâncias e pontos dentro da área formada pelo campo de visão da câmera. Em ambas configurações, o sistema consegue identificar e reproduzir as respostas vibratórias apropriadas para as condições “Perto” e “Longe” respectivamente. Isso permitiu completar a divisão do campo de visão da câmera em seis setores: S1, S2 e S3 para a condição “Longe”, e S4, S5 e S6 para a condição “Perto” (Figura 23).

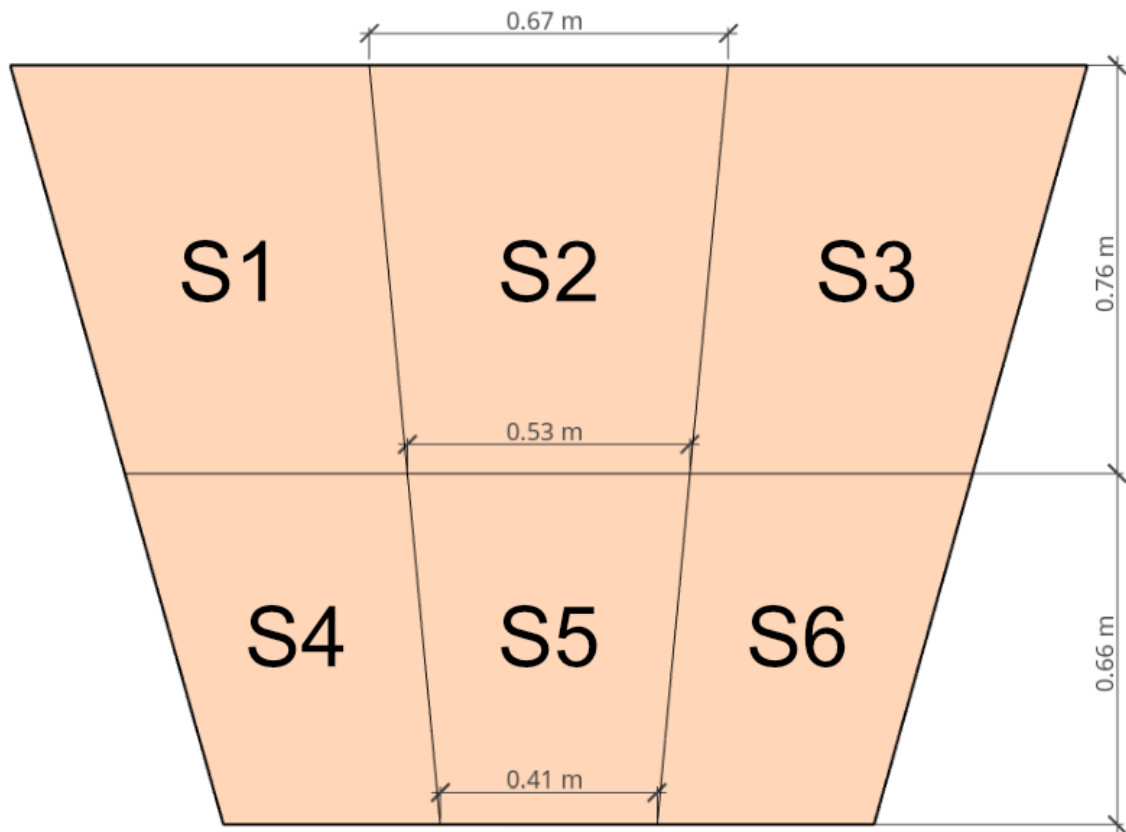


Figura 23. Divisão e dimensão dos setores no campo de visão da câmera. Imagem criada pelo autor.

A verificação do funcionamento do sistema foi analisada de forma diferente para cada situação. Para a condição “Longe”, foi considerada mais importante a capacidade do sistema identificar e emitir uma primeira resposta em tempo suficiente, enquanto para a condição “Perto” o mais importante é a capacidade do sistema se manter emitindo respostas enquanto o código QR se mantiver dentro do campo de visão da câmera. O sistema, então, se comportou conforme o esperado, conforme é possível perceber na figura 24, onde a taxa de identificação dentro dos setores da condição “Perto” é consistentemente alta e compatível com o comportamento do valor controle, enquanto que a taxa e identificação nos setores da situação “Longe”, apesar de apresentarem maior desvio padrão entre si e de terem menor número absoluto de identificações, são capazes de identificação em tempo adequado e emitir o primeiro alerta ao usuário.

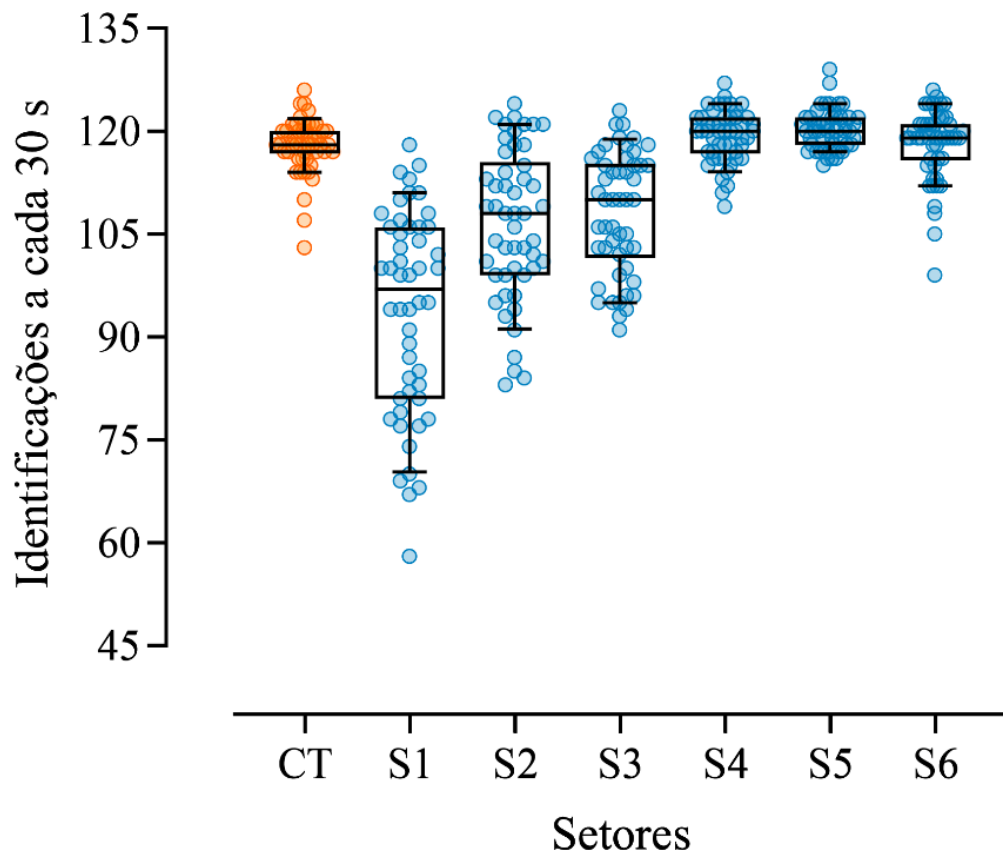


Figura 24. Taxa de identificação por setor, comparados à situação controle (CT). Imagem criada pelo autor.

Por fim, foi realizado teste acerca de como o sistema se comporta em variados níveis de intensidade luminosa no ambiente. O experimento foi feito em sala sem luz e sem janelas, para que a única fonte de luz fosse uma lâmpada tipo *spot* halógena, com intensidade luminosa máxima de 1000 cd, erguida a 2,35 m do chão, direcionada ao código QR “O” no chão. A altura da câmera em relação ao piso foi de 1,2 m, com o campo de visão direcionado de modo que o código QR se localizasse no centro da imagem, no limite do setor S5. A lâmpada foi ligada através de um *dimmer* para que fosse possível ajuste gradual de tensão, assim possibilitando diversos níveis de intensidade luminosa (figura 25).



Figura 25. Configuração para o teste de iluminação. Imagem registrada pelo autor.

O nível mais baixo de intensidade luminosa para que fosse possível detecção de código QR foi de 58 cd, no qual a capacidade de identificação do código QR foi extremamente instável, porém totalmente abaixo em comparação ao controle. A estabilidade da precisão aumentou conforme a intensidade luminosa alcançou 71 cd, onde foi possível ver quantidades mais próximas de identificação a cada teste de 30 s de duração, alcançando estabilidade superior ao CT a partir de 140 cd de intensidade luminosa, onde o tal nível de precisão se manteve até a saída de tensão máxima da lâmpada, conforme ilustrado no gráfico da figura 26. Concluimos que isso pode ter acontecido devido à parte branca do papel ofuscar a tinta preta usada para imprimir o código QR, ofuscamento este possivelmente provocado pela escolha de uma resolução de vídeo mais baixa (que diminui a nitidez) ou por causa da própria capacidade da câmera.

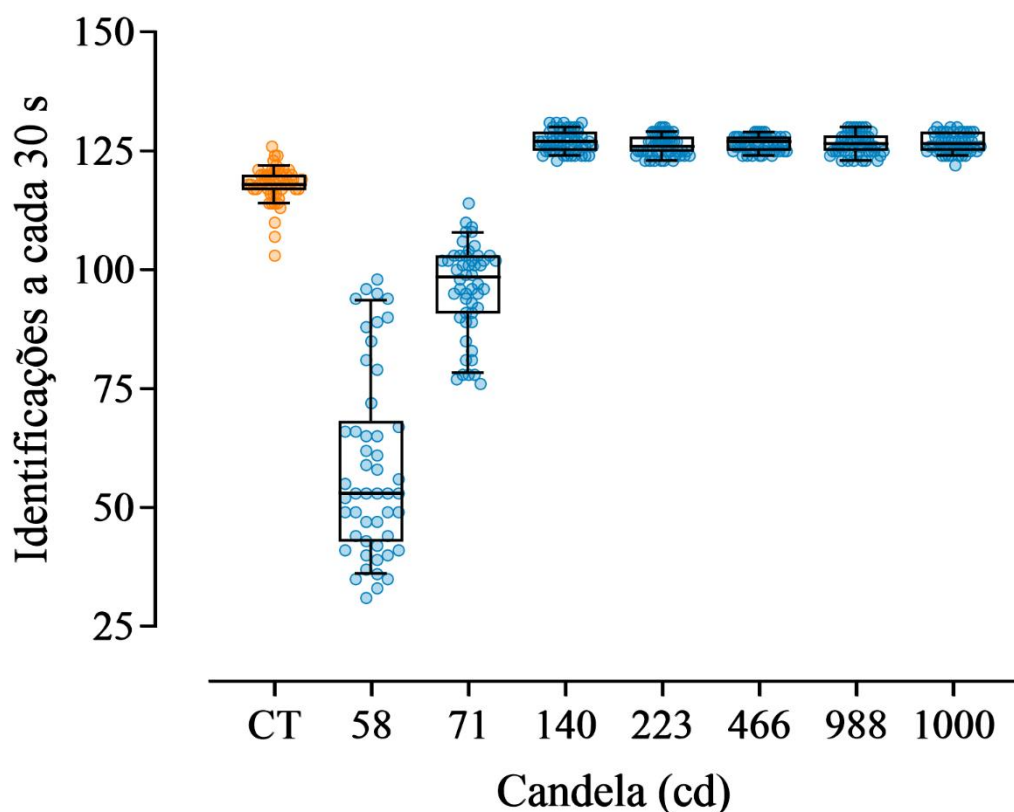


Figura 26. Taxa de identificação por nível de iluminação, comparados à situação controle (CT). Imagem criada pelo autor.

5.2 TESTE MÓVEL EM CORREDOR

Uma vez finalizados os testes referentes ao funcionamento do objeto de maneira fixa, foi necessário desenvolver uma forma de analisar se continuariam adequados com o sistema em

movimento. Para tal, além de organizar um modo para que o sistema fosse testado em movimento ao mesmo tempo em que fosse monitorado, teve que ser organizado um espaço para que fossem realizados os testes.

Para a montagem do sistema em formato móvel e estável, sem interferência humana direta que alterasse altura ou inclinação (exemplo: segurar o objeto com as mãos, prender o objeto em vestimentas, usar o objeto sem qualquer tipo de suporte adequado), foi utilizada uma mesa de dois níveis com rodas (figura 27). No nível superior da mesa, foram posicionados o sistema na altura planejada em relação ao chão (1,2 m), uma tela, um teclado e um mouse. No nível inferior foi fixada uma extensão de



Figura 27. Mesa utilizada para os testes móveis, com o sistema, monitor, teclado e mouse. Imagem registrada e editada pelo autor.

tomada de 2 m para a tela e para testes comparativos sem e com bateria conectada. Uma segunda extensão de tomada de 10 m foi utilizada para permitir maior alcance de movimento.

O local escolhido foi um corredor de 22,85 m de comprimento e de 2,30 m de largura. Em virtude do monitoramento com tela e do espaçamento das tomadas, foram feitos dois conjuntos de medições: primeiramente foram feitos testes de monitoramento com tela em um trecho de 13,50 m de comprimento. E uma vez feitos os ajustes necessários, foram realizados testes em um segundo trecho de 11,15 m para avaliar o sistema ligado somente na bateria e sem tela, ou seja, totalmente independente, da forma que deve ser operado quando finalizado. Em ambos trechos haviam barreiras próprias do corredor (pilares) e móveis de uso público (bancos e lixeiras), assim como objetos oriundos dos laboratórios acadêmicos, que também foram organizados de maneira a formar obstáculos.

No primeiro conjunto de testes, em virtude do uso de uma tela para monitoramento, utilizou-se o sistema ligado à tomada para evitar que a bateria compromettesse o processamento de alguma forma, uma vez que o intuito era testar: a orientação e distanciamento entre os códigos QR (assim como seus posicionamentos em relação ao sistema), o *delay* das resoluções

e filtros de imagem escolhidos, e se houve permanência de respostas atrasadas. No segundo conjunto de testes, com o sistema ligado somente na bateria, avaliou-se a velocidade de identificação, consistência e precisão das respostas emitidas, múltiplos códigos QR simultâneos, e, por fim, funcionamento contínuo do sistema.

No primeiro conjunto de testes, feito no trecho de 13,50 m (figura 28), a orientação dos códigos QR foi uniformizada: os sinais visuais foram dispostos de frente para a câmera. Os códigos “C”, para indicar “caminho à frente”, foram posicionados no centro do corredor. Os pilares e móveis do corredor foram utilizados como “obstáculos” e sinalizados através do código QR “O” nos cantos externos mais próximos do sentido do trajeto. Por fim, o código QR “F”, para indicar “final do caminho”, foi posicionado no centro, logo à frente da parede do fim do corredor. Também foi estudado o posicionamento dos códigos QR nos objetos em si, mas a angulação da câmera fez com que esses códigos fossem detectados muito próximos ao sistema, aumentando a possibilidade de choques mecânicos. Deste modo, preferiu-se mantê-los dispostos no piso.

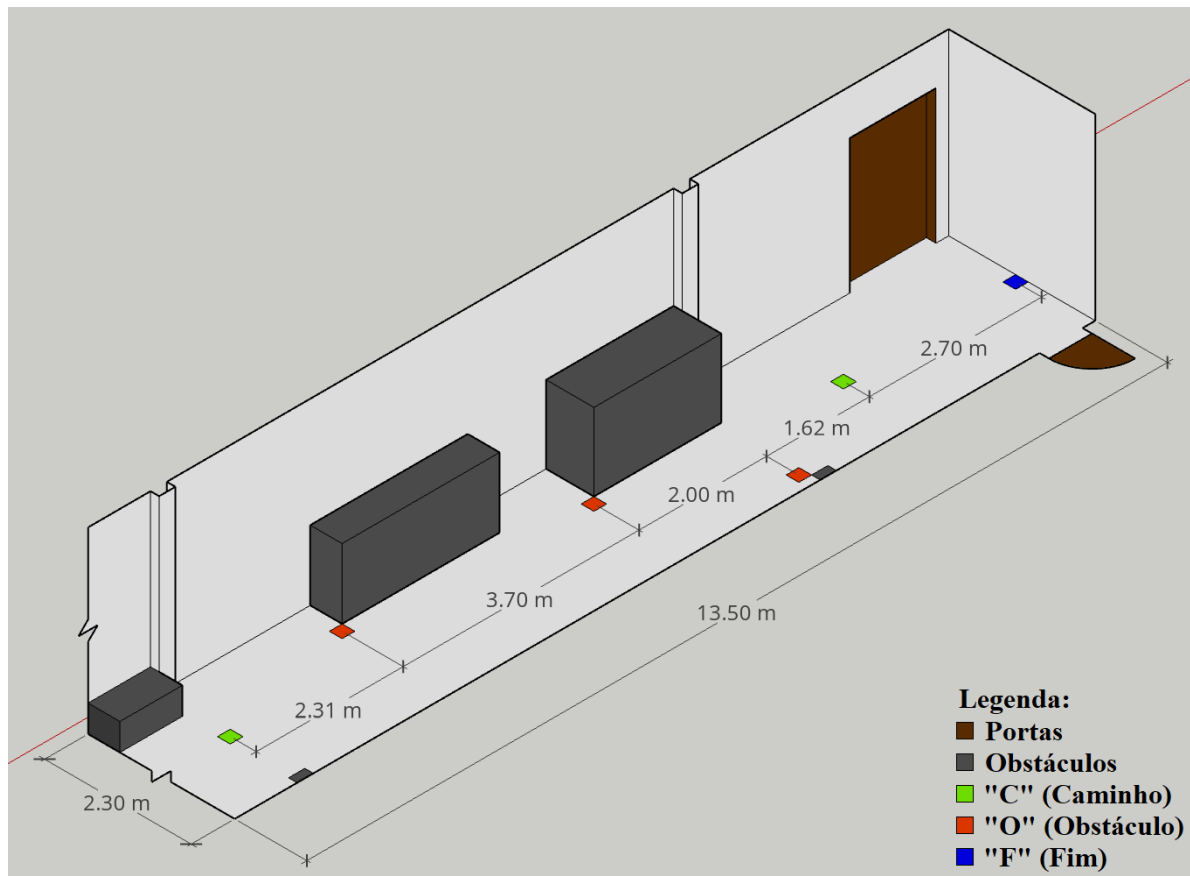


Figura 28. Simulação tridimensional cotada do trecho do corredor utilizado no primeiro conjunto de testes móveis. Imagem criada pelo autor.

As distâncias entre os códigos QR variaram entre 1 a 4 m nesta fase de testes. Isso foi assim feito para monitorar a permanência de emissão de respostas ou atraso de processamento de novas respostas após um determinado código já ter saído das imagens captadas pelo sistema devido a possíveis *delays*. Isso permitiu a verificação de que na resolução em 480p 4:3 houve um *delay* significativo com o sistema em movimento, a ponto de o sistema continuar a emitir respostas vibratórias por cerca de 1 a 2 s após os códigos QR terem saído do campo de visão da câmera. Por outro lado, a resolução de 360p 4:3 apresentou *delay* abaixo de 1 s, não gerando a mesma emissão contínua após a saída do código QR das imagens da câmera. Ao mesmo tempo, com a troca de resoluções, foi verificado maior número de detecções bem-sucedidas na resolução 360p 4:3, devido a menor tamanho das informações a serem processadas pelo sistema, acelerando a emissão de novas respostas. Não houve perda significativa de qualidade a ponto de afetar a identificação dos códigos QR. Com base nesse resultado, pôde-se diminuir as distâncias entre os códigos QR no segundo conjunto de testes.

Para finalizar a primeira etapa de testes móveis, foi conectada a bateria ao sistema. Aqui foram feitos trajetos de ida e volta, para verificar se a bateria seria capaz de suprir as necessidades do objeto em movimento. Os resultados foram favoráveis e não houve qualquer tipo de atraso, permanência de respostas ou falhas na identificação dos códigos QR. O corredor também apresentou variados níveis de intensidade luminosa, o que reforçou a capacidade apropriada de identificação do dispositivo conforme o teste de iluminação (figura 29).

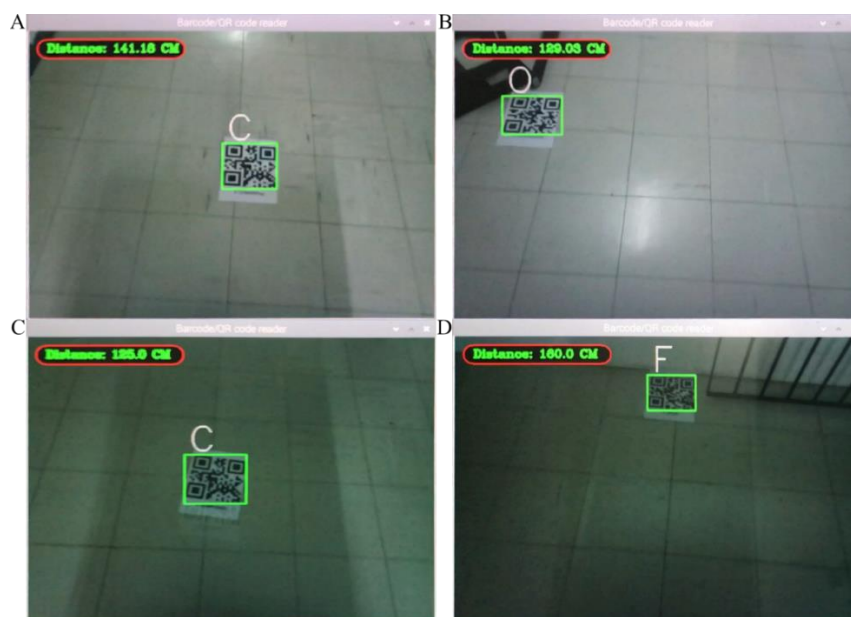


Figura 29. Demonstração da capacidade apropriada de identificação do dispositivo no que se refere aos variados níveis de intensidade luminosa presentes no corredor. Imagens registrada pelo autor.

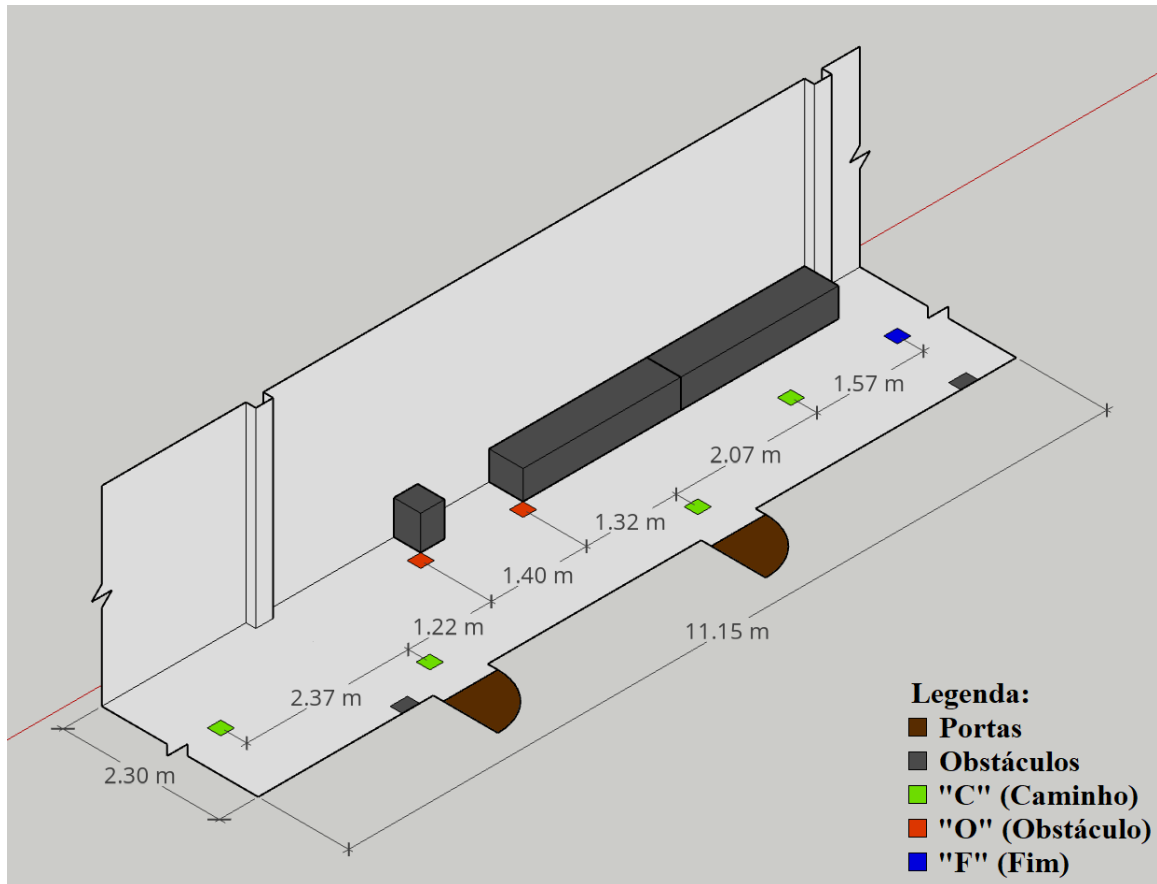


Figura 30. Simulação tridimensional cotada do trecho do corredor utilizado no segundo conjunto de testes móveis. Imagem criada pelo autor.

No segundo conjunto de testes, feito no trecho de 11,15 m (figura 30), o sistema, já ligado na bateria, foi testado inicialmente com auxílio de tela para monitoramento (figura 31), a fim de capturar o modo como se comportava perante situações de múltiplos códigos QR, códigos rotacionados e mais próximos um do outro. Neste trecho, a orientação da maioria dos códigos QR dispostos foi de frente para a câmera, com três códigos posicionados de lado em relação ao sentido da câmera. Dois destes foram códigos “C” em frente das duas portas na lateral direita, e um código “O”



Figura 31. Sistema ligado na bateria, com tela para monitoramento das imagens. Imagem registrada e editada pelo autor.

na lateral esquerda, para indicar uma lixeira grande. Por fim, o código QR “F”, para indicar “final do caminho” foi posicionado no centro do corredor à frente da parede no segundo trecho foi colocado aonde o primeiro trecho começava. Nesta fase de testes, as distâncias entre os códigos foram encurtadas, variando entre 1 a 2,5 m, com base nos resultados de *delay* de respostas adquiridos durante os primeiros testes no corredor.

Com auxílio da tela, assegurou-se que o sistema funcionou como planejado e emitiu as respostas vibratórias conforme esperado (figura 32-A), independente de orientação (figura 32-B), e conforme as distâncias adequadas (figura 33), até mesmo em quando foram identificados códigos simultâneos, acionando respostas para o sinal visual mais próximo até este sair das imagens, para então emitir as respostas referentes ao mais longe (figura 34).

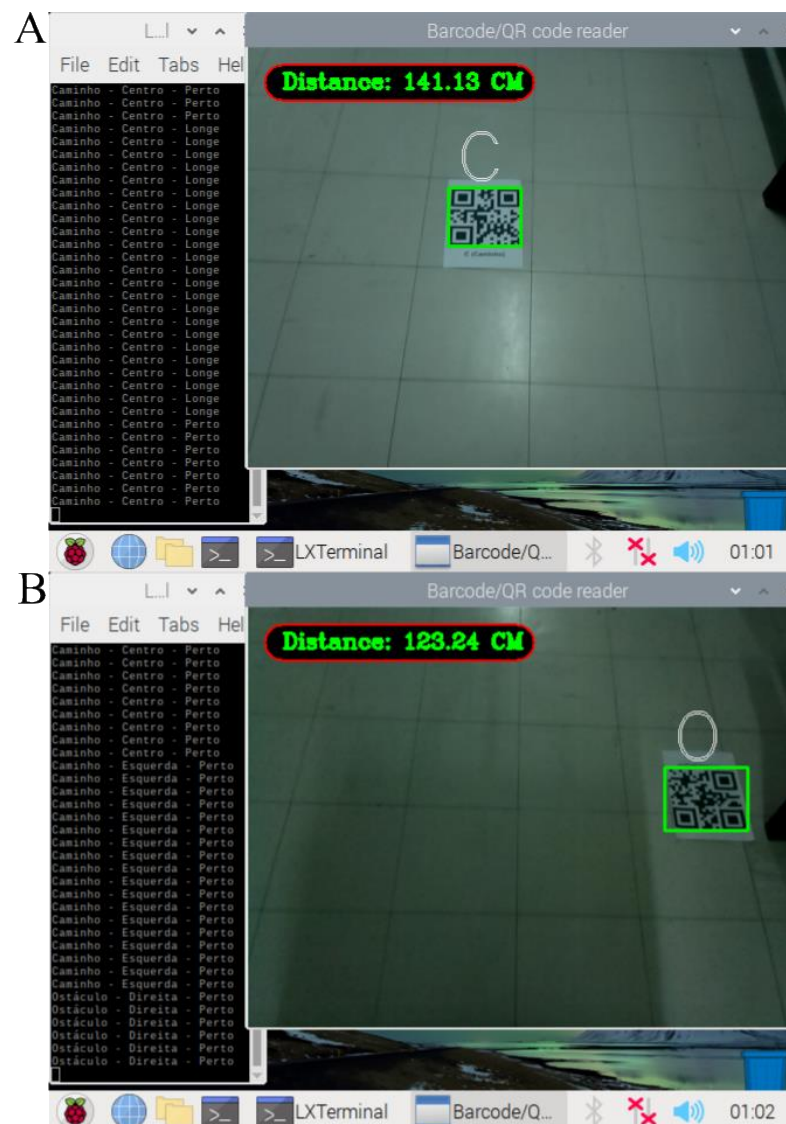


Figura 32. Sistema identificando um código QR em situação padrão (A) e rotacionado em 180° (B). Imagens registradas pelo autor.

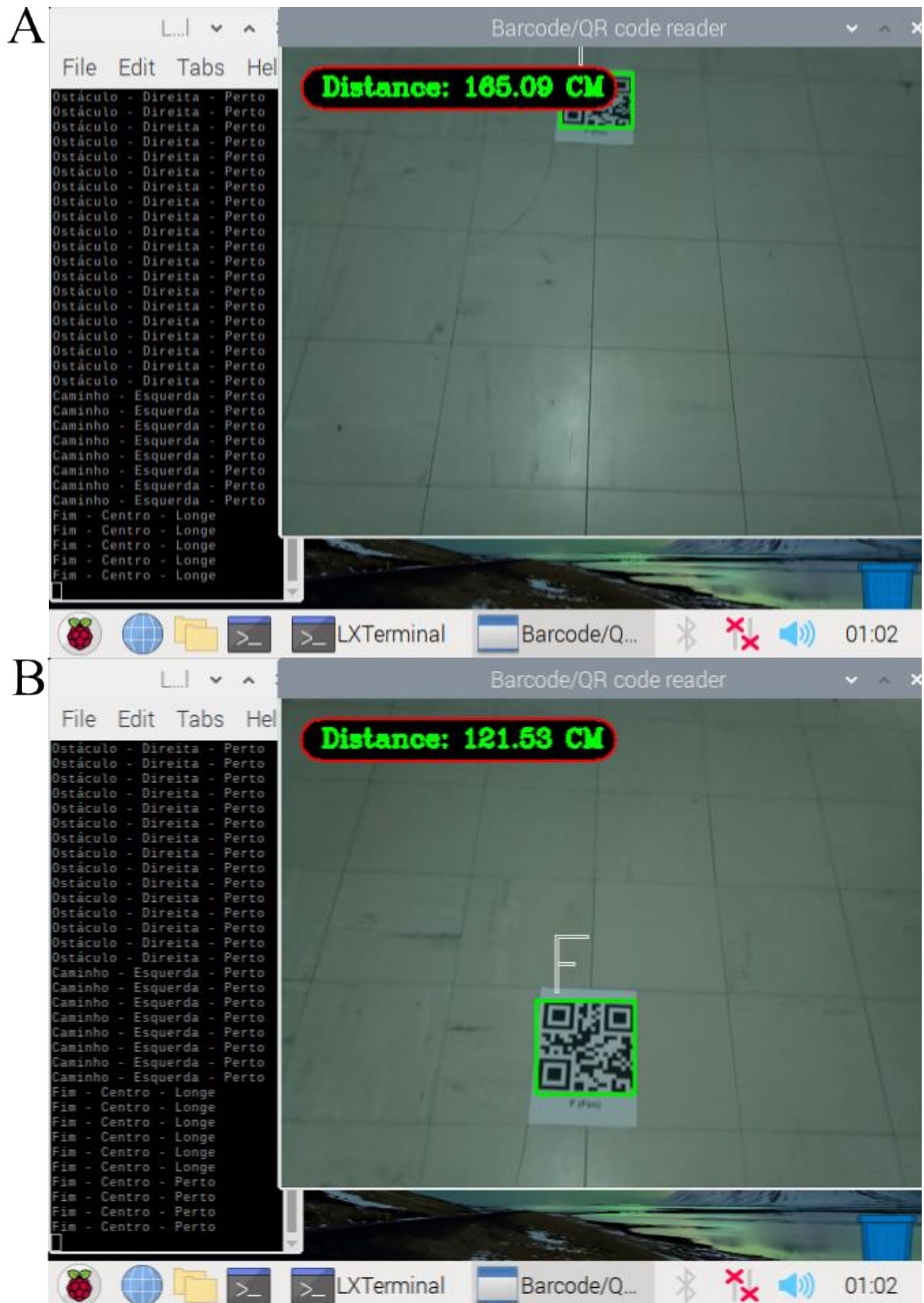


Figura 33. Sistema identificando um código QR em diferentes distâncias. Imagens registradas pelo autor.

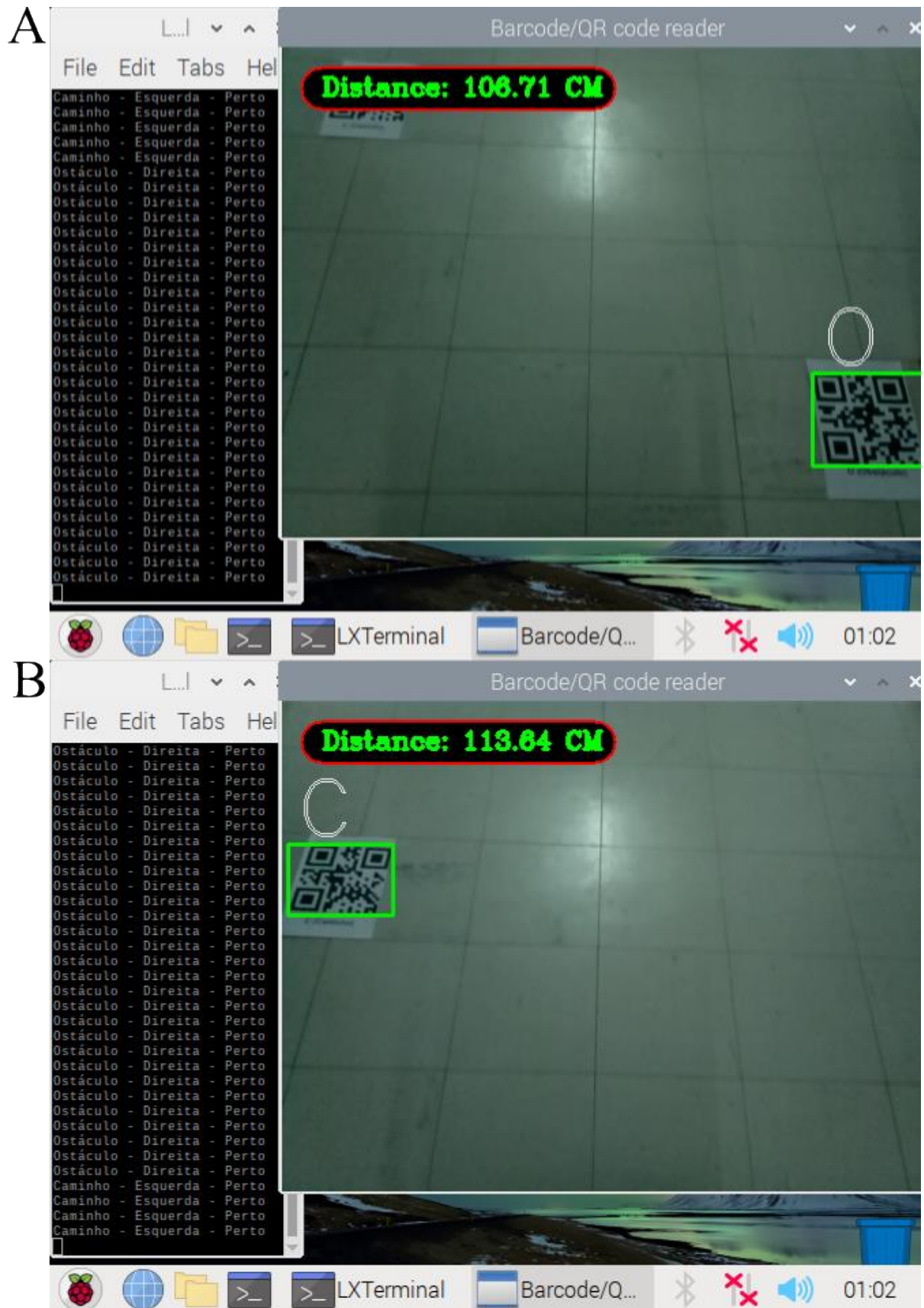


Figura 34. Sistema identificando códigos QRs simultâneos. Imagens registradas pelo autor.

Uma vez verificado o trecho da segunda fase de testes pelo monitoramento com tela, o sistema foi testado como foi planejado para ser usado: sem tela, ligado na bateria e somente emitindo respostas através dos motores (figura 35). O trecho de 11,15 m foi percorrido a uma velocidade de aproximadamente 0,4 m/s (1,44 km/h), velocidade abaixo do caminhar de uma pessoa sem deficiência visual (CLARK-CARTER, 1986). Nesta velocidade de andar, o sistema foi capaz de identificar com sucesso os códigos QR e alertar através dos motores, tanto em termos de orientação quanto de padrões de vibrações, que se provaram suficientemente distintos um do outro. O mesmo percurso foi realizado com velocidades mais altas, a aproximadamente 0,7 m/s (2,5 km/h), para testar a eficácia do sistema nessas condições, e o sistema ainda se provou eficiente na leitura das informações.



Figura 35. Teste de funcionamento do sistema sem tela e ligado à bateria, à velocidade de caminhar de aproximadamente 0,4 m/s (1,44 km/h). Imagem registrada e editada pelo autor.

Uma vez finalizados os testes de funcionamento do dispositivo, foi adicionada emissão de respostas em áudio para complementar as respostas táteis e tornar o resultado mais claro ao usuário. Esse tipo de emissão é opcional para uso, uma vez que é necessário colocar fone de ouvido para que se ouça as respostas, já que não foi instalado no sistema módulo para emissão de áudio próprio. Em seu estado atual, as respostas também somente se referem ao tipo de código QR identificado: “*Obstáculo*”, “*Caminho*” e “*Fim*”, cabendo aos motores táteis a informação em respeito à direção e proximidade. No entanto, se o usuário preferir, as respostas podem ser adicionadas ou trocadas conforme necessidades apresentadas. As respostas em áudio são emitidas a cada 15 ciclos no código, ou seja, a cada 15 emissões de respostas táteis, uma resposta em áudio é emitida. Isso foi assim realizado para evitar o acúmulo excessivo de informações dentro da memória do sistema, que pode causar atrasos e até mesmo erros a ponto de fechar o programa para liberar espaço na memória. Para o usuário, isso será percebido como uma resposta auditiva a cada 4 s, aproximadamente.

5.3 PROTÓTIPO

Com o objeto funcionando nos moldes desejados, foi necessário rever o modelo do objeto finalizado para atender às necessidades reformuladas: que possa ser vestido facilmente, que seja ajustável conforme o tamanho do usuário, que possa se firmar no corpo do usuário sem ficar mudando de posição constantemente e que atenda às exigências previamente estabelecidas de altura, inclinação e posicionamento das partes do sistema.

A primeira revisão do objeto finalizado envolveu a troca do colete por outro tipo de acessório. Isso foi escolhido em virtude do possível desconforto em relação a temperatura (principalmente em ambientes quentes) como pela maior dificuldade em vestir o objeto, se comparado com outros métodos levados em consideração posteriormente. Outro fator considerado foi o possível superaquecimento das peças do sistema, em virtude de provável falta de ventilação do tecido do colete. Os métodos então considerados envolveram o uso de alças ou cintos, eventualmente levando ao modelo de alças de mochilas que possam ser ajustadas tanto por trás quanto pela frente, no qual pudesse ser acoplado o sistema sem que ele fosse obstruído por tecido de qualquer forma.

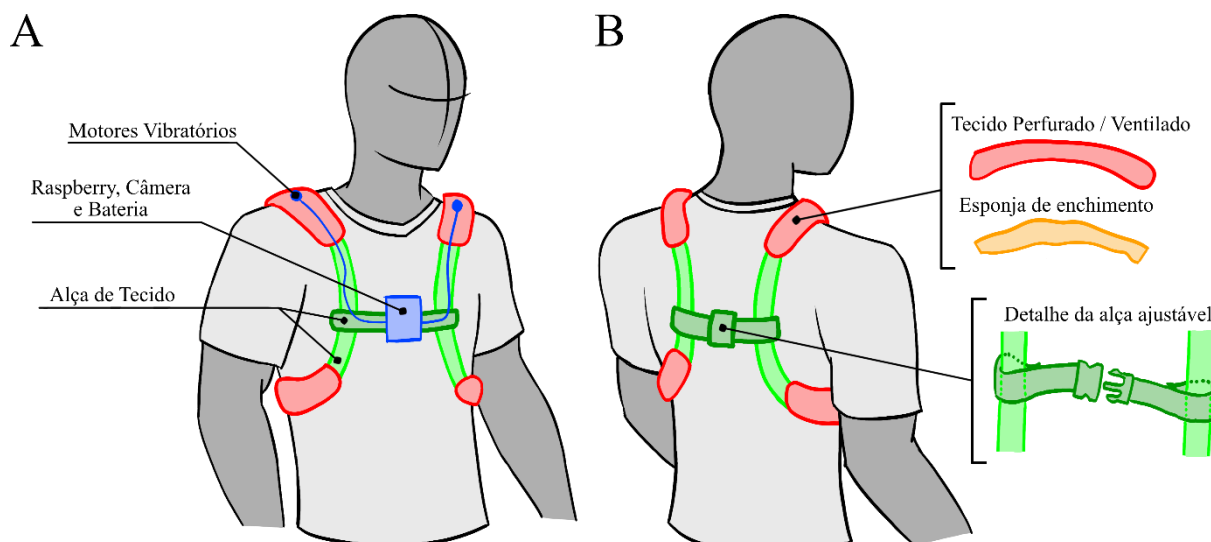


Figura 36. Esquema atualizado, com vista frontal (A) e dorsal (B). Imagem criada pelo autor.

Seguindo as novas necessidades observadas, foi esquematizado um novo desenho para o objeto (figura 36). O novo suporte para o objeto de utiliza de alças de tecido para suportar o sistema como um alças de uma mochila. A altura da câmera pode ser ajustada verticalmente conforme as necessidades do usuário, assim como as tiras de tecido podem ser ajustadas para se firmarem melhor ao corpo. Como alternativa para facilitar o uso do suporte, pode ser incluso um fecho na alça dorsal para ajudar o usuário a fixar mais o objeto em si. Para impedir que o suporte se movimente ou fique fora de foco conforme o usuário se movimenta, escolheu-se utilizar enchimento de esponja nas regiões do ombro e costelas para segurar o suporte na posição intendida, para permitir maior superfície de contato sem trazer incômodo ao usuário.

Com esse modelo esquematizado, foi trabalhado como ficaria o posicionamento das partes do sistema nas alças. A placa Raspberry e a câmera seriam posicionadas no centro, na parte frontal do suporte, através de um case que se encaixe na alça. A posição dos motores nos ombros se manteve, mas os cabos que os conectam à placa foram reposicionados de modo a seguirem a orientação das tiras de tecido, seja por costura ou por alças às quais podem ser passados por dentro. Estudou-se a posição da bateria portátil na parte dorsal do objeto, mas por fim foi decidido não a posicionar no suporte para minimizar o peso e não causar comprometimentos significativos com o equilíbrio geral do sistema junto ao usuário, uma vez que os suportes dos ombros e das costelas impedem que o objeto se mova significativamente. Conforme estas mudanças, foram feitos ajustes no modelo real das alças (figura 37).



Figura 37. Imagens das alças confeccionadas para o sistema de frente (A), de costas (B), em perfil (C, D), e detalhes dos encaixes e ajustes (D). Imagens registradas pelo autor.

No modelo final, foram adicionadas mais duas alças horizontais, para estabilizar mais a movimentação das alças no corpo, assim como foram adicionadas mais três áreas de ajustes e encaixes, todos posicionados agora na parte frontal das alças para melhor acesso por parte do usuário. O modelo da caixa de proteção (figura 38) possui 125,5 *milímetros* (mm) de altura, 76,8 mm de largura e 37,5 mm de profundidade, com demais detalhamentos dos buracos para passagem de cabos e ventilação descritos no Apêndice B. Já impresso, o modelo é dividido em três peças, presas uma à outra por parafusos de 2 mm de diâmetro: a base da Raspberry Pi e da câmera (por onde também é fixado às alças do suporte), a caixa com buracos para passagem de cabos e ventilação, assim como para a câmera, e a porta com ventilação para acesso às portas de entrada da Raspberry Pi (figuras 39 e 40). Escolheu-se a alça horizontal superior frontal para a Raspberry Pi, diminuindo extensão de cabos e não afetando de maneira significativa o equilíbrio do dispositivo durante uso. A posição dos motores se manteve nas alças dos ombros, onde foram fixados na parte interior para melhor detecção dos sinais vibratórios (figura 41).

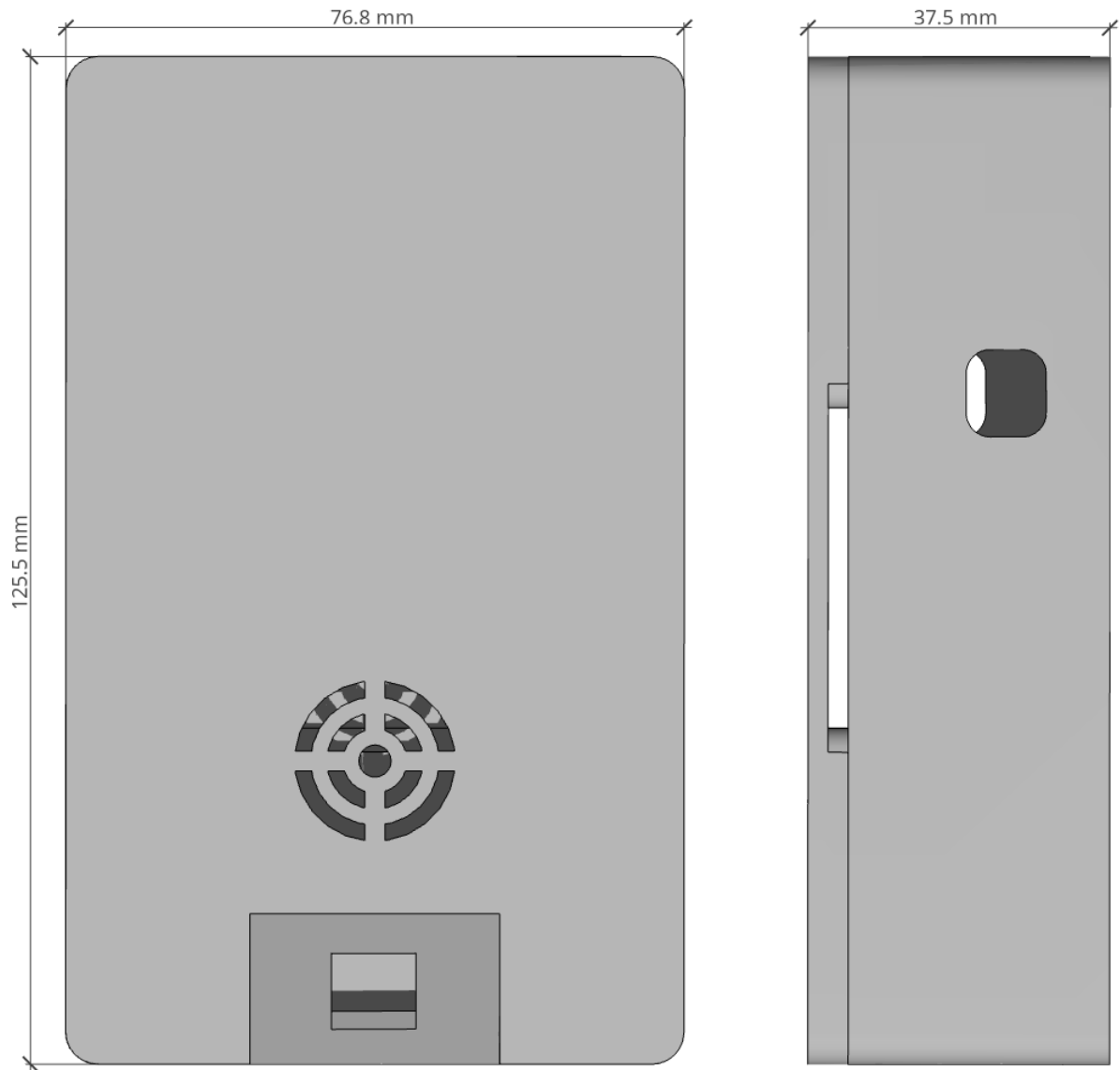


Figura 38. Dimensões finais do modelo da caixa a ser impressa. Imagem criada pelo autor.

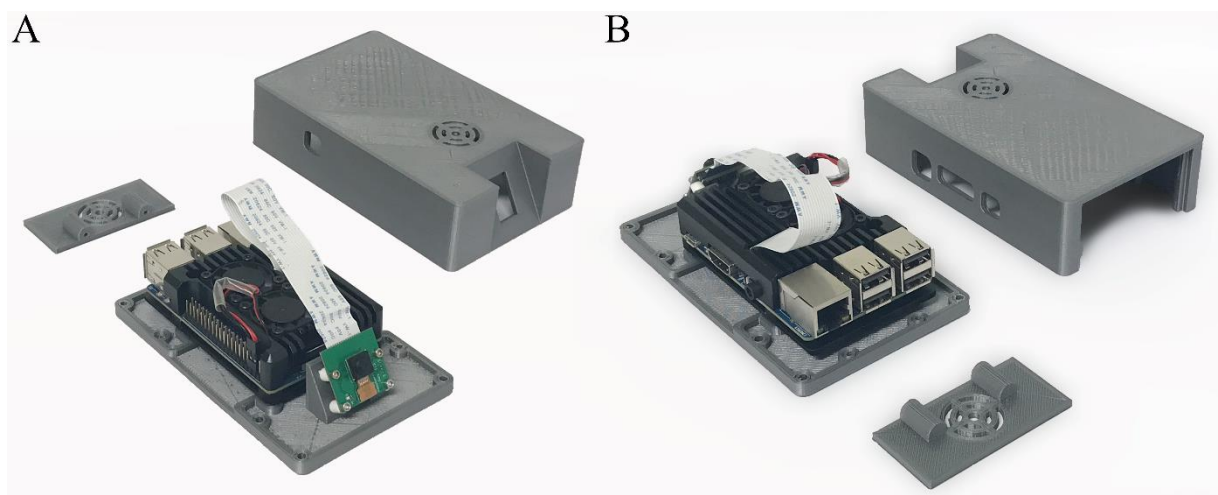


Figura 39. Imagens das peças de impressão 3D em vistas diferentes (A: inferior direita. B: superior esquerda). Imagens registradas pelo autor.

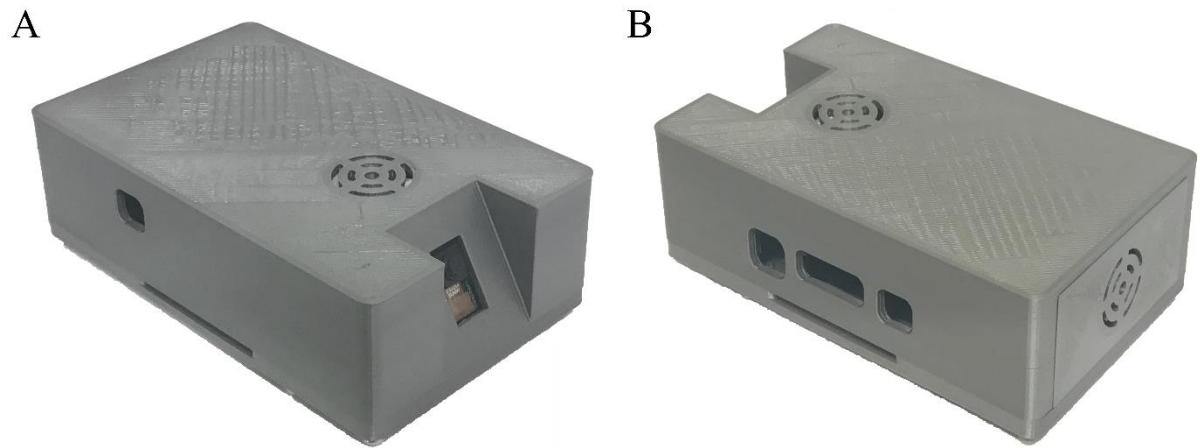


Figura 40. Imagens do modelo de impressão 3D montado em vistas diferentes (A: inferior direita. B: superior esquerda). Imagens registradas pelo autor.

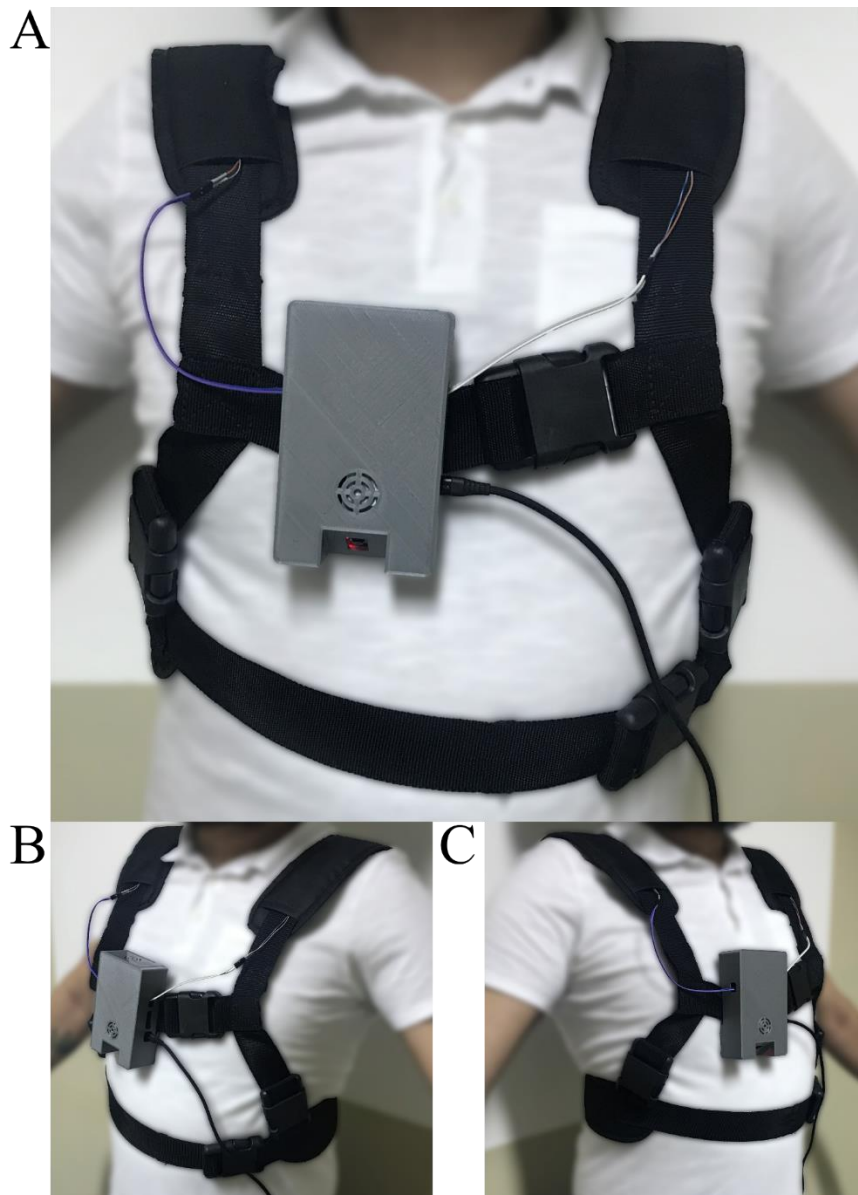


Figura 41. Imagens do protótipo finalizado (A, B, C). Imagens registradas pelo autor.

6 CONCLUSÃO

O sistema desenvolvido neste trabalho preza pelo zelo a pessoas dentro de seus ambientes privativos, e para tal encontrou no uso de aparatos tecnológicos uma forma de fazer tal tarefa. Permitir a alguém se locomover com segurança dentro de seu ambiente mais particular, sem riscos a si ou a seus pertences, foi o intuito durante a criação deste projeto.

Os resultados aqui obtidos comprovam a capacidade de identificar obstáculos a distâncias de até 1,4 m à frente com clareza e alertar ao usuário através de respostas não invasivas, porém perceptíveis. Uma vez que haja compreensão dos padrões vibratórios e do que representam, é possível que uma pessoa possa utilizar-se deste sistema, ou de uma versão aprimorada deste, para se orientar dentro de sua residência. A simplicidade do sistema operacional também fornece fácil manutenção e personalização de comandos por parte de técnicos, podendo até mesmo ser adicionado novos padrões de vibração através de novos códigos específicos.

Ainda que opcionais, foi observado que complementar as respostas vibratórias com mensagens em áudio demonstrou aumentar a precisão da resposta emitida, como é notado em sistemas que também possuem mais de um tipo de respostas, como o trabalho de WACHAJA et al (2017), o de PATIL et al (2018) e o de KLEINBERG et al (2023), ou arranjo complexo de respostas do mesmo tipo, como de LEE e IN (2023). Portanto, considerou-se adequada a combinação de tipos de respostas no dispositivo aqui criado, uma vez que possui respostas principais através de vibrações, que normalmente são mais aceitas pelos usuários conforme descrito em MADAKE et al (2023), e respostas complementares em mensagens em áudio através de fones de ouvido, que fornecem mais detalhes e melhor orientação se não forem emitidas em número elevado como considerado em TAPU et al (2020).

O sistema aqui desenvolvido, apesar de menos capaz em termos de computação de dados quando comparado a sistemas de *smartphones* como em JEON et al (2012), TAPU et al (2013) ou CHACCOUR e BADR (2015), ou outros dispositivos que se utilizam de auxílio de inteligências artificiais ou sistemas de processamento remoto, tem a vantagem de ser simples e autônomo. Não estar conectado à internet, *Bluetooth* ou a outras formas de comunicação sem fio permite que o aparelho não sofra influências que comprometam seu funcionamento, uma vez que captura as imagens, processa os dados e emite respostas por si mesmo. Ler um código QR não-cadastrado também não oferece riscos, uma vez que o código não executa nenhuma informação adquirida, mas transcreve seu conteúdo para um arquivo “.txt” e compara com as respostas cadastradas, imediatamente apagando se houver ou não correspondência.

Ademais, como a plataforma Raspberry Pi é amplamente comercializada a baixo custo, o sistema oferece maior liberdade ao usuário, não limitando o usuário a peças personalizadas ou *hardwares* especialmente feitos. Qualquer uma de suas peças pode ser trocada por outra de diferente qualidade, custo ou função, como a troca da câmera por uma de infravermelho, para que o sistema possa identificar códigos QR em ambientes mal iluminados, ou trocar a bateria por outra caso a primeira descarregue ou mesmo alterar de fato por outra de maior capacidade ou mais leve. A própria placa Raspberry Pi 3 pode ser substituída por um modelo menor, como a placa Raspberry Pi Zero 2 W, ou um mais potente, como a Raspberry Pi 4, como usada em KLEINBERG et al (2023). Com quaisquer substituições, o sistema ainda possuiria tamanho pequeno e peso leve o suficiente para não atrapalhar o usuário significativamente. Porém, há de se considerar que o dispositivo aqui desenvolvido é voltado para orientação autônoma, não sendo adequado nem capaz de guiar ou escolher os melhores caminhos para o usuário, como visto em sistemas feitos para ativamente ajudar a locomover o usuário, como os vistos em WACHAJA et al (2015, 2017) e JIMÉNEZ et al (2020).

No momento do desenvolvimento deste trabalho, o custo total do projeto é de aproximadamente R\$ 1.500,00 (mil e quinhentos reais). Esse valor é mais voltado para baixo custo se comparado a sistemas de computação mais complexa, porém há de se considerar que sistemas que se utilizam de aplicativos para celulares são mais acessíveis se o usuário já possui um *smartphone* compatível. No entanto, considera-se também que, se o objetivo desses aplicativos é ativamente orientar o usuário, isso impediria o usuário de usar seu celular para usos pessoais, tais como ligações e leitura em voz alta de mensagens, portanto avaliou-se que para tais sistemas o uso de um dispositivo adicional, neste caso outro *smartphone*, é necessário em comparação ao sistema aqui desenvolvido.

Outro fator que auxilia a longevidade do sistema é o código aberto do *software*, que permite personalização facilitada dos comandos, como novos padrões de vibração e até mesmo novas mensagens em áudio. Manter o sistema em código aberto também o que impede sua obsolescência por ser disponibilizado ao público para manutenção, correção de erros, instalação de aprimoramentos e adição de novas funções.

Desta maneira, o projeto aqui desenvolvido é capaz de identificar com sucesso os códigos visuais criados em distância e tempo adequados, assim como emitir com precisão respostas vibratórias através dos motores e auditivas através dos fones de ouvido. Em seu estado atual, pode ser facilmente adaptado conforme as necessidades particulares do usuário graças ao seu algoritmo altamente personalizável.

7 TRABALHOS FUTUROS

No campo de funcionalidades em softwares, é possível adicionar novas respostas junto às vibrações e áudio, como palavras e frases inteiras para identificação de ambientes e obstáculos específicos, suspensos ou desnivelados. A aplicação de inteligência artificial para reconhecer padrões visuais poderia necessitar de hardware mais potentes, tais como placas Raspberry Pi mais atuais, minicomputadores ou *smartphones*, mas pode ser considerada uma possibilidade de expansão se considerarmos diminuição no custo de comercialização desses artigos e haja a criação um banco de dados devidamente alimentado com as situações desejadas.

Outra forma de expansão do sistema é através da comunicação com outros hardwares, como expandir as formas de identificação através de sistemas auxiliares que utilizem ondas rádio ou comunicação wireless entre si. Esses outros aparelhos seriam posicionados nos ambientes desejados e alertariam o sistema principal quando este chegasse próximo demais a eles, indicando possível choque. Para tal, seria necessário adicionar instrumentos medidores posicionados nas paredes, para determinar limites físicos mais precisos dentro da residência do usuário, recriando os ambientes em software dentro do aparelho principal.

Finalmente, é possível uma expansão do sistema para a esfera urbana e possibilitar o uso do produto em ambientes públicos e externos, utilizando-se de aparatos urbanísticos já existentes, como o piso tátil, postes de iluminação e placas de sinalização, para o posicionamento dos sinais visuais adequados, em materiais apropriados para maior durabilidade. Com a ajuda das respostas auditivas, o grande número de objetos e obstáculos típicos do ambiente urbano podem ser cadastrados e permitir ao usuário maior autonomia em sua vida pública.

REFERÊNCIAS

ABDON, LEONARDO MIRANDA. *Tecnologia assistiva na arquitetura das smart homes: o uso de sensores para aplicação no ambiente interno de instituições de longa permanência para idosos*. Faculdade de Arquitetura e Urbanismo, Universidade Federal do Pará, 2019.

Associação Brasileira de Normas Técnicas (ABNT). *ABNT NBR 16537 – Acessibilidade – Sinalização tátil no piso – Diretrizes para elaboração de projetos e instalação*. ABNT, Primeira Edição, 2016.

Associação Brasileira de Normas Técnicas (ABNT). *ABNT NBR ISO/CIE 89950-1 – Iluminação de ambientes de trabalho – Parte 1: Interior*. ABNT, 2013.

Associação Brasileira de Normas Técnicas (ABNT). *ABNT NBR 9050 – Acessibilidade a edificações, mobiliário, espaços e equipamentos urbanos*. ABNT, Terceira Edição, 2015.

B. S. TJAN, P. J. BECKMANN, R. ROY, N. GIUDICE AND G. E. LEGGE. *Digital Sign System for Indoor Wayfinding for the Visually Impaired*. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) – Workshops*, pp. 30-30. San Diego, CA, USA, 2005.

BATALLA, J. M., VASILAKOS, A., & GAJEWSKI, M.. *Secure Smart Homes*. *ACM Computing Surveys*, 50(5), 1–32. doi:10.1145/3122816. 2017.

BERSCH, Rita. *INTRODUÇÃO À TECNOLOGIA ASSISTIVA*. In *Assistiva: Tecnologia e Educação*, 2017. Disponível em: <<http://www.assistiva.com.br/tassistiva.html>>. Acesso em 23 de Abril de 2023.

BRASIL – SDHPR – Comitê de Ajudas Técnicas – ATA VII. 2007. Link para acesso: <https://www.assistiva.com.br/Ata_VII_Reunião_do_Comite_de_Ajudas_Técnicas.pdf>. Acesso em 23 de Abril de 2023.

Camera. In *Raspberry Pi Documentation*. Link para acesso: <<https://www.raspberrypi.com/documentation/accessories/camera.html>>. Acessado em: 20 de Junho de 2022.

CHACCOUR, K., & BADR, G. *Novel indoor navigation system for visually impaired and blind people*. In *2015 International Conference on Applied Research in Computer Science and Engineering (ICAR)*. (pp. 1– 5). Piscataway, NJ: IEEE. 2015.

CLARK-CARTER, D. D., HEYES, A. D., & HOWARTH, C. I.. *The efficiency and walking speed of visually impaired people*. Ergonomics, 29(6), 779–789. doi:10.1080/00140138608968314. 1986.

DAL, ASADULLAH. *Distance Estimation Using Single Camera OpenCV Python*. In *LinkedIn*. Link para acesso: <https://www.linkedin.com/pulse/distance-estimation-using-single-camera-opencv-python-asadullah-dal?trk=public_profile_article_view>. Acessado em 16 de Agosto de 2022.

DE BELEN, R. A. J., BEDNARZ, T., & FAVERO, D. D.. *Integrating Mixed Reality and Internet of Things as an Assistive Technology for Elderly People Living in a Smart Home*. The 17th International Conference on Virtual-Reality Continuum and Its Applications in Industry on - VRCAI '19. doi:10.1145/3359997.3365742. 2019.

FIGUEIRO, MARIANA GROSS. *A proposed 24 h lighting scheme for older adults*. Lighting research and technology, v. 40, n. 2, p 153 - 160, 2008. In SOUSA, Isabella Gaspar; MAIA, Ivana Márcia Oliveira. *Arquitetura de interiores em ambientes para idosos portadores da doença de Alzheimer*. Arq.Urb , v. 11, p. 192-207, 2014.

Five Ways To Run a Program On Your Raspberry Pi At Startup. In *DEXTER Industries*. Link para acesso: <<https://www.dexterindustries.com/howto/run-a-program-on-your-raspberry-pi-at-startup/>>. Acessado em 3 de Outubro de 2022.

GUVEN, BEHIC. *Building a Barcode/QR code Reader using Python*. In *Towards Data Science*, 2020. Link para acesso: <<https://towardsdatascience.com/building-a-barcode-qr-code-reader-using-python-360e22dfb6e5>>. Acessado em: 28 de Julho de 2021.

JEON, M. et al. *"Listen2dRoom": helping blind individuals understand room layouts*. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*. Texas, USA, 2012.

JIMÉNEZ, M. F., MELLO, R. C., BASTOS, T., & FRIZERA, A.. *Assistive Locomotion Device with Haptic Feedback For Guiding Visually Impaired People*. Medical Engineering & Physics. doi: 10.1016/j.medengphy.2020.04.002. 2020.

KLEINBERG, D., YOZEVITCH, R., ABEKASIS, I., ISRAEL, Y. e HOLDENGREBER, E.. *A Haptic Feedback System for Spatial Orientation in the Visually Impaired: A Comprehensive Approach*. In *IEEE Sensors Letters*, vol. 7, no. 9, pp. 1-4, Sept. 2023, Art no. 6005804, doi: 10.1109/LENS.2023.3307068.

LEE, M. e IN, H.. *Novel Wrist-Worn Vibrotactile Device for Providing Multi-Categorical Information for Orientation and Mobility of the Blind and Visually Impaired*. In *IEEE Access*, vol. 11, pp. 111860-111874, 2023, doi: 10.1109/ACCESS.2023.3321802.

MADAKE, J., BHATLAWANDE, S., SOLANKE, A. e SHILASKAR, S.. *A Qualitative and Quantitative Analysis of Research in Mobility Technologies for Visually Impaired People*. In IEEE Access, vol. 11, pp. 82496-82520, 2023, doi: 10.1109/ACCESS.2023.3291074.

MADAKE, J., BHATLAWANDE, S., SOLANKE, A. e SHILASKAR, S.. *PerceptGuide: A Perception Driven Assistive Mobility Aid Based on Self-Attention and Multi-Scale Feature Fusion*. In IEEE Access, vol. 11, pp. 101167-101182, 2023, doi: 10.1109/ACCESS.2023.3314702.

MARIKYAN, D., PAPAGIANNIDIS, S., & ALAMANOS, E.. *A systematic review of the smart home literature: A user perspective*. Technological Forecasting and Social Change. doi: 10.1016/j.techfore.2018.08.015. 2018.

NCD Risk Factor Collaboration (NCD-RisC). *A century of trends in adult human height*. In eLife, 5, doi:10.7554/elife.13410, 2016.

On Raspberry Pi - Auto start terminal after login?. In Stack Overflow. Link para acesso: <<https://stackoverflow.com/questions/36466500/on-raspberry-pi-auto-start-terminal-after-login>>. Fórum publicado em 7 de Abril de 2016. Acessado em 3 de Outubro de 2022.

PAIVA, S., & GUPTA, N.. *Technologies and Systems to Improve Mobility of Visually Impaired People: A State of the Art*. EAI/Springer Innovations in Communication and Computing, 105–123. doi:10.1007/978-3-030-16450-8_5. 2019

PATIL, K., JAWADWALA, Q., & SHU, F. C.. *Design and Construction of Electronic Aid for Visually Impaired People*. IEEE Transactions on Human-Machine Systems, 48(2), 172–182. doi:10.1109/thms.2018.2799588. 2018.

PUBLIC LAW 100-407-AUG. 19, 1988. *Technology related assistance for individuals with disabilities act*. Estados Unidos, 1988.

PUBLIC LAW 105-394-NOV. 13, 1998. *Assistive Technology Act of 1998*. Estados Unidos, 1998.

PULS, MAURÍCIO MATTOS. *Arquitetura e Filosofia*. São Paulo: Annablume, 2006. Disponível em: <https://books.google.com.br/books?id=HK_iYLYwWOIC&printsec=frontcover&hl=pt-BR#v=onepage&q&f=false>. Acessado em 23 de Abril de 2023.

Raspberry Pi Hardware. In *Raspberry Pi Documentation*. Link para acesso: <<https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>>. Acessado em: 20 de Junho de 2022.

SANTOSO, F. AND S. J. REDMOND. *Indoor location-aware medical systems for smart homecare and telehealth monitoring: state-of-the-art*. In *Physiological Measurement*, Vol. 36, No. 10, pp. 53– 87. 2015.

SCHIMPL, M., MOORE, C., LEDERER, C., NEUHAUS, A., SAMBROOK, J., DANESH, J., ... DAUMER, M.. *Association between Walking Speed and Age in Healthy, Free-Living Individuals Using Mobile Accelerometry—A Cross-Sectional Study*. PLoS ONE, 6(8), e23299, 2011.

SOUSA, ISABELLA GASPAR; MAIA, IVANA MÁRCIA OLIVEIRA. *Arquitetura de interiores em ambientes para idosos portadores da doença de Alzheimer*. Arq.Urb , v. 11, p. 192-207, 2014.

TAPU, R., MOCANU, B., & ZAHARIA, T.. *Wearable assistive devices for visually impaired: A state of the art survey*. Pattern Recognition Letters. doi:10.1016/j.patrec.2018.10.031. 2020.

TAPU, R., MOCANU, B., BURSUC, A., & ZAHARIA, T.. *A Smartphone-Based Obstacle Detection and Classification System for Assisting Visually Impaired People*. 2013 IEEE International Conference on Computer Vision Workshops. doi:10.1109/iccvw.2013.65. 2013.

TEPELEA, L. et al. *A Vision Module for Visually Impaired People by Using Raspberry PI Platform*. 2019 15th International Conference on Engineering of Modern Electric Systems (EMES). 2019: 209-212.

WACHAJA, A. et al. *Navigating blind people with a smart walker*. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, 2015, pp. 6014-6019.

WACHAJA, A. et al. *Navigating blind people with walking impairments using a smart walker*. In Springer Link, 2017. Link para acesso: <<https://link.springer.com/article/10.1007/s10514-016-9595-8#citeas>>. Acessado em: 23 de Setembro de 2019.

World Health Organization. *9D90 Vision impairment including blindness*. In International statistical classification of diseases and related health problems (11th ed.). 2019. Link para acesso: <<https://icd.who.int/browse11/l-m/en#/http%3a%2f%2fid.who.int%2fid%2fentity%2f1103667651>>. Acessado em: 26 de Setembro de 2022.

World Health Organization. *Universal Eye Health: A Global Action Plan 2014-2019*. Geneva, Switzerland: WHO; 2013. Link para acesso:

<<https://www.who.int/publications/i/item/universal-eye-health-a-global-action-plan-2014-2019>>. Acessado em: 26 de Setembro de 2022.

ZAFAR, S. et al.. *Assistive Devices Analysis for Visually Impaired Persons: A Review on Taxonomy*. IEEE Access, vol. 10, pp. 13354-13366. 2022.

APÊNDICES

APÊNDICE A – CÓDIGO EM PYTHON

0. Bibliotecas importadas

0.1. Câmera

import cv2

0.2. QRs

from pyzbar import pyzbar

0.3. Intervalos e criação de padrões

from time import sleep, time

0.4. Motores

from gpiozero import PWMOutputDevice

0.5. Áudio

import vlc

1. Variáveis básicas

1.1. Variável para ativação de respostas em áudio

flag_audio = 0

1.2. Valor referente à distância real, usado no cálculo da distância

Known_distance = 50

1.3. Valor referente ao tamanho real do código QR, usado no cálculo da distância

Known_width = 19.5

1.4. Proporção para adequar o tamanho e distância real para o virtual (com o código QR alinhado)

ref_qr_width = 175

settings = dict(qr_width = 0, qr_center = 0)

1.5. Cores

GREEN = (0, 255, 0)

RED = (0, 0, 255)

WHITE = (255, 255, 255)

BLACK = (0, 0, 0)

1.6. Fonte

fonts = cv2.FONT_HERSHEY_COMPLEX

1.7. Detector de código QR

qr_detector = cv2.QRCodeDetector()

1.8. Função para verificar distância focal

def Focal_Length_Finder(measured_distance, real_width, width_in_rf_image):

 # Para achar "focal_length"

 focal_length = (width_in_rf_image * measured_distance) / real_width

 return focal_length

1.9. Função para estimar distância

def Distance_finder(Focal_Length, real_face_width, face_width_in_frame):

 distance = (real_face_width * Focal_Length) / face_width_in_frame

 # Retorna "distance"

 return distance

1.10. Função para leitura do QR:

def read_barcodes(frame):

 # 1.10.1. Definindo width:

 global width

 # 1.10.2. Decodificador do código QR:

 barcodes = pyzbar.decode(frame)

 # 1.10.3. Detalhamento do código QR no "frame":

 for barcode in barcodes:

1.10.3.1. Medição e localização do código QR em pixels:

x, y, w, h = barcode.rect

1.10.3.2. Definir equivalência entre a largura real em CM e a largura virtual em pixels:

settings['qr_width'] = w

settings['qr_center'] = x + (w/2)

1.10.3.3. Formar o quadrado ao redor do código QR:

barcode_info = barcode.data.decode('utf-8')

cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 237, 0), 2)

1.10.3.4. Identificação do QR e representação do texto:

font = cv2.FONT_HERSHEY_DUPLEX

cv2.putText(frame, barcode_info, (x + 6, y - 6), font, 2.0, (237, 237, 237), 1)

1.10.3.5. Armazenamento da informação em um arquivo .txt:

with open("/home/lapa/Desktop/barcode_result.txt", mode='w') as file:

file.write(barcode_info)

return frame

2. Cálculo para equivalência no tamanho aparente no frame:

Focal_length_found = Focal_Length_Finder(Known_distance, Known_width, ref_qr_width)

3. Motores:

motor1 = PWMOutputDevice(13) # -> Motor Esquerdo

motor2 = PWMOutputDevice(12) # -> Motor Direito

4. Câmera:

4.1. Criação da janela "frame":

camera = cv2.VideoCapture(0)

camera.set(3, 480)

camera.set(4, 360)

ret, frame = camera.read()

4.2. Loop da Câmera

while ret:

4.2.1. Criação do frame:

ret, frame = camera.read()

frame = cv2.rotate(frame, cv2.ROTATE_180)

frame = read_barcode(frame)

4.2.2. Checar status do "qr_width":

#print(settings['qr_width'])

4.2.3. Checar se "qr_width" não é igual a 0, ou seja, se o QR está ou não no frame. Se sim, ativa a sequência de if's.

if settings['qr_width'] != 0:

4.2.3.1. Acha a distância através da função Distance finder:

Distance = Distance_finder(Focal_length_found, Known_width, settings['qr_width'])

4.2.3.2. Faz a legenda que diz a distância calculada e o sombreamento para fácil leitura no "frame":

cv2.line(frame, (30, 30), (230, 30), RED, 32)

cv2.line(frame, (30, 30), (230, 30), BLACK, 28)

cv2.putText(frame, f"Distance: {round(Distance, 2)} CM", (30, 35), fonts, 0.6, GREEN, 2)

4.2.4. Vídeo colorido:

cv2.imshow('Barcode/QR code reader', frame)

4.2.5. Vídeo em Preto e Branco, sem tons de cinza:

#grayFrame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

blackAndWhiteFrame = cv2.threshold(grayFrame, 100, 255, cv2.THRESH_BINARY)

#cv2.imshow('video bw', blackAndWhiteFrame)

4.2.6. Resultado em tempo real da leitura do código QR:

Obs.: Limpar o arquivo .txt antes do loop (usar somente em casos de erros de leitura do .txt):

```
#open('barcode_result.txt', 'w').close()
```

```
with open('/home/lapa/Desktop/barcode_result.txt', 'r') as file:
```

```
    qr = file.read().replace("\n", "")
```

4.2.6.1. QR de "Obstáculo":

```
if qr == 'O':
```

```
    # Contagem do flag_audio para emissão de resposta em áudio:
```

```
    flag_audio = flag_audio+1
```

```
    # Toca o áudio "Obstáculo":
```

```
    if flag_audio == 15:
```

```
        print('Áudio')
```

```
        p = vlc.MediaPlayer('/home/lapa/Desktop/Obstáculo.mp3')
```

```
        p.play()
```

```
        flag_audio = 0
```

```
        sleep(1)
```

```
        p.stop()
```

```
    # Se o código QR estiver posicionado à esquerda do vídeo:
```

```
    if settings['qr_center'] < 160:
```

```
        # Se a distância verificada for abaixo ou igual a 150 CM:
```

```
        if Distance <= 150:
```

```
            print('Obstáculo - Esquerda - Perto')
```

```
            motor1.value = 1
```

```
            sleep(.15)
```

```
        # Se a distância verificada for acima de 150 CM:
```

```
        else:
```

```
            print('Obstáculo - Esquerda - Longe')
```

```
            motor1.value = 0.5
```

```
            sleep(.15)
```

```
    # Se o código QR estiver posicionado à direita do vídeo:
```

```

elif settings['qr_center'] > 320:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Ostáculo - Direita - Perto')
        motor2.value = 1
        sleep(.15)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Ostáculo - Direita - Longe')
        motor2.value = 0.5
        sleep(.15)
# Se o código QR estiver posicionado no centro do vídeo:
else:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Ostáculo - Centro - Perto')
        motor1.value = 1
        motor2.value = 1
        sleep(.15)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Ostáculo - Centro - Longe')
        motor1.value = 0.5
        motor2.value = 0.5
        sleep(.15)

```

4.2.6.2. QR de "Caminho":

```

elif qr == 'C':
    # Contagem do flag_audio para emissão de resposta em áudio:
    flag_audio = flag_audio + 1
    # Toca o áudio "Caminho":
    if flag_audio == 15:
        print('Áudio')
        p = vlc.MediaPlayer('/home/lapa/Desktop/Caminho.mp3')

```

```

p.play()
flag_audio = 0
sleep(1)
p.stop()

# Se o código QR estiver posicionado à esquerda do vídeo:
if settings['qr_center'] < 160:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Caminho - Esquerda - Perto')
        motor1.value = 1
        sleep(.05)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Caminho - Esquerda - Longe')
        motor1.value = 0.5
        sleep(.05)

# Se o código QR estiver posicionado à direita do vídeo:
elif settings['qr_center'] > 320:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Caminho - Direita - Perto')
        motor2.value = 1
        sleep(.05)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Caminho - Direita - Longe')
        motor2.value = 0.5
        sleep(.05)

# Se o código QR estiver posicionado no centro do vídeo:
else:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Caminho - Centro - Perto')
        motor1.value = 1

```

```

        motor2.value = 1
        sleep(.05)
# Se a distância verificada for acima de 140 CM:
else:
    print('Caminho - Centro - Longe')
    motor1.value = 0.5
    motor2.value = 0.5
    sleep(.05)

# 4.2.6.3. QR de "Final do Caminho":
elif qr == 'F':
    # Contagem do flag_audio para emissão de resposta em áudio:
    flag_audio = flag_audio + 1
    # Toca o áudio "Fim":
    if flag_audio == 15:
        print('Áudio')
        p = vlc.MediaPlayer('/home/lapa/Desktop/Fim.mp3')
        p.play()
        flag_audio = 0
        sleep(1)
        p.stop()
    # Se o código QR estiver posicionado à esquerda do vídeo:
    if settings['qr_center'] < 160:
        # Se a distância verificada for abaixo ou igual a 150 CM:
        if Distance <= 150:
            print('Fim - Esquerda - Perto')
            motor1.value = 0.5
            sleep(.25)
        # Se a distância verificada for acima de 150 CM:
        else:
            print('Fim - Esquerda - Longe')
            motor1.value = 0.25
            sleep(.25)
    # Se o código QR estiver posicionado à direita do vídeo:

```



```

elif settings['qr_center'] > 320:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Fim - Direita - Perto')
        motor2.value = 0.5
        sleep(.25)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Fim - Direita - Longe')
        motor2.value = 0.25
        sleep(.25)
# Se o código QR estiver posicionado no centro do vídeo:
else:
    # Se a distância verificada for abaixo ou igual a 150 CM:
    if Distance <= 150:
        print('Fim - Centro - Perto')
        motor1.value = 0.5
        motor2.value = 0.5
        sleep(.25)
    # Se a distância verificada for acima de 150 CM:
    else:
        print('Fim - Centro - Longe')
        motor1.value = 0.25
        motor2.value = 0.25
        sleep(.25)

```

4.2.7. Limpa os valores dos motores antes de reiniciar o loop.:

```

motor1.value = 0
motor2.value = 0

```

4.2.8. Limpa o .txt antes de reiniciar o loop:

```

open('/home/lapa/Desktop/barcode_result.txt', 'w').close()

```

4.2.9. Pressionar "Esc" para fechar a câmera:

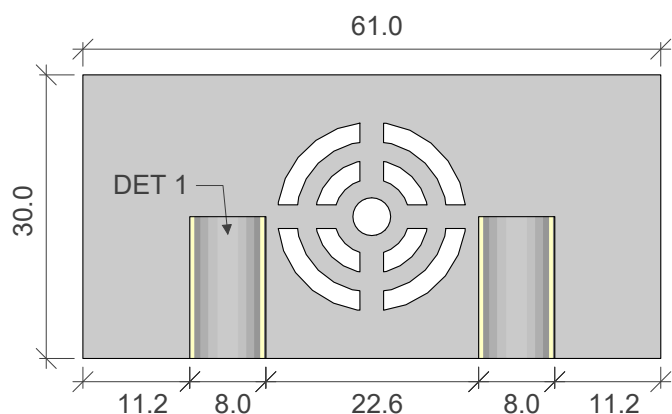
```
if cv2.waitKey(1) & 0xFF == 27:  
    break
```

4.3. Interrompe o loop do código após o "break" da câmera:

```
camera.release()  
cv2.destroyAllWindows()
```

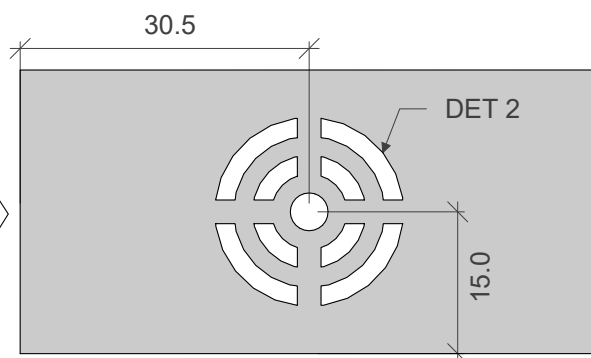
APÊNDICE B – DETALHAMENTO DA CAIXA DE PROTEÇÃO

Segue em anexo nas páginas seguintes as pranchas técnicas (com medidas em milímetros) das peças que compõem o modelo desenhado para a caixa de proteção do projeto.



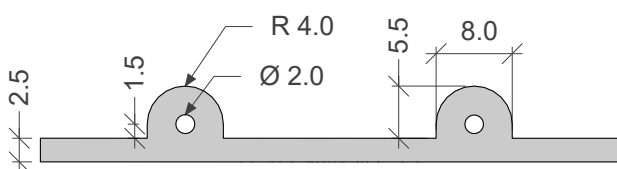
01

VISTA FRONTAL DA PORTA DE ACESSO
ESCALA: 1.25:1 EM MM



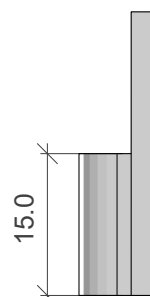
02

VISTA TRASEIRA DA PORTA DE ACESSO
ESCALA: 1.25:1 EM MM



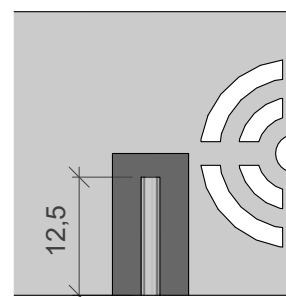
03

VISTA INFERIOR DA PORTA DE ACESSO
ESCALA: 1.25:1 EM MM



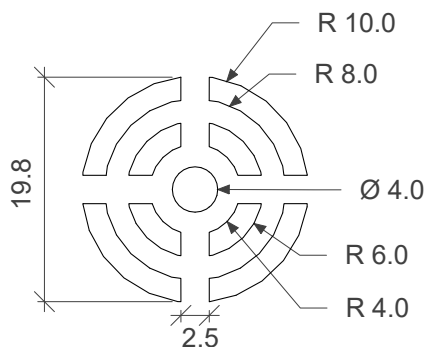
04

VISTA 1 - VISTA LATERAL
ESCALA: 1.25:1 EM MM



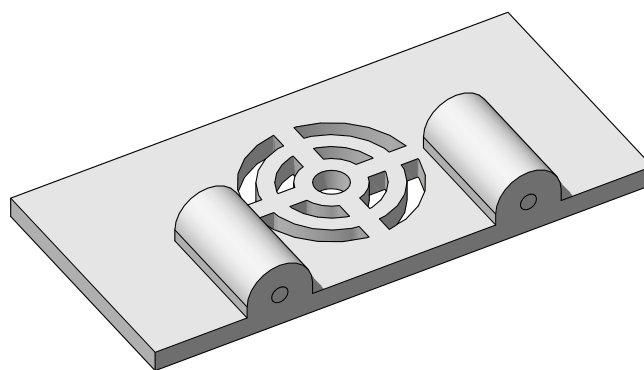
05

DETALHE 1 - FURO ROSCADO
ESCALA: 1.25:1 EM MM



06

DETALHE 2 - ABERTURA DE VENTILAÇÃO
ESCALA: 1.5:1 EM MM



07

VISTA ISOMÉTRICA DA PORTA DE ACESSO
ESCALA: 1.25:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

DISCENTE:
LEONARDO MIRANDA ABDON

Nº DE MATRÍCULA:
202017470002

CONTEÚDO:
PORTA DE ACESSO ÀS ENTRADAS USB

PRANCHA
Nº

ORIENTADOR:
MANOEL DA SILVA FILHO

ESCALA:
INDICADA

INSTITUTO DE CIÊNCIAS BIOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E
BIOLOGIA CELULAR PPGNBC

1

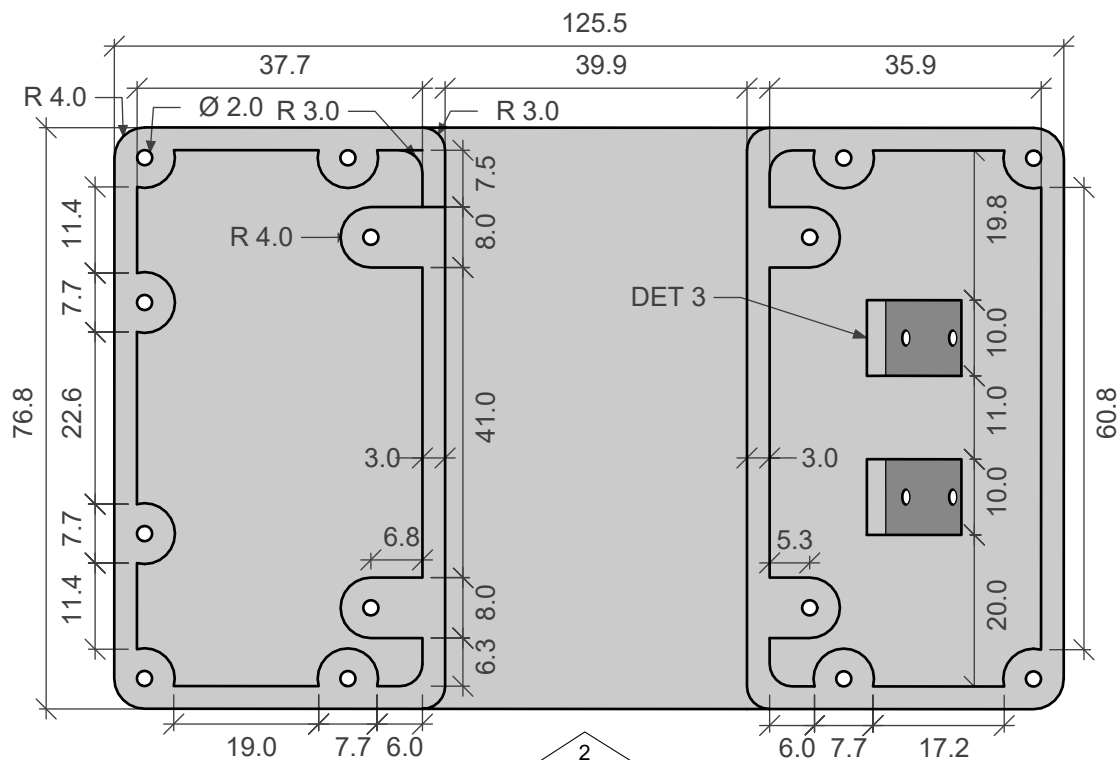
LABORATÓRIO:
LABORATÓRIO DE PROTOTIPAGEM
ASSISTIVA - LAPA

DATA:
NOV 2023

DISSERTAÇÃO DE MESTRADO

9

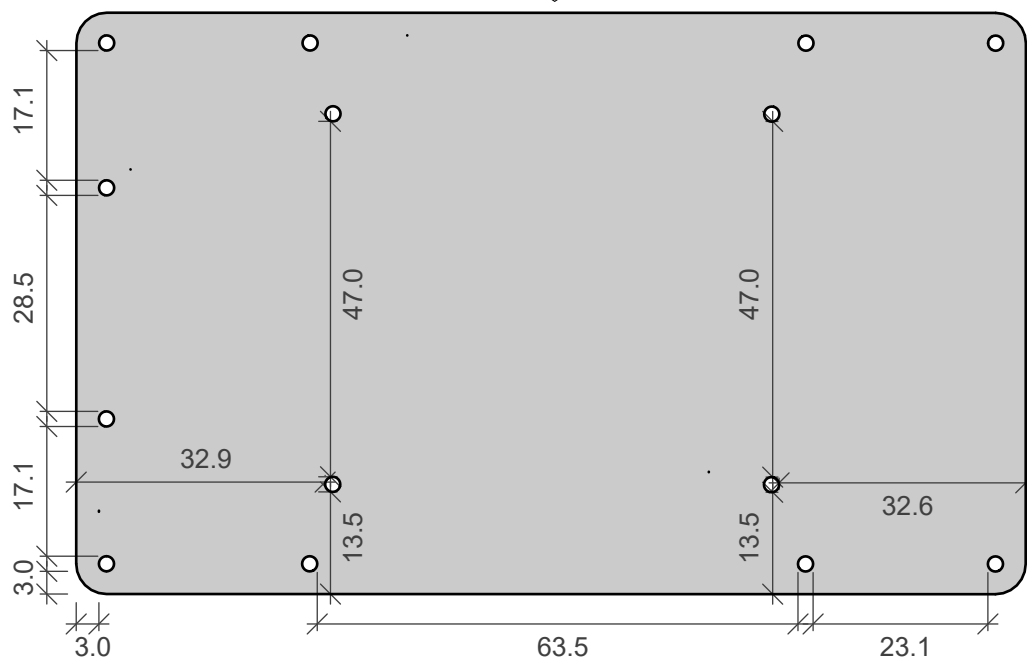
TEMA:
MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"



08

VISTA SUPERIOR DA BASE

ESCALA: 1:1 EM MM



09

VISTA INFERIOR DA BASE

ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

DISCENTE:
LEONARDO MIRANDA ABDON

Nº DE MATRÍCULA:
202017470002

CONTEÚDO:
BASE DA CAIXA PROTETORA DA PLACA RASPBERRY PI

PRANCHA
Nº

ORIENTADOR:
MANOEL DA SILVA FILHO

ESCALA:
INDICADA

INSTITUTO DE CIÊNCIAS BIOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E
BIOLOGIA CELULAR PPGNBC

2

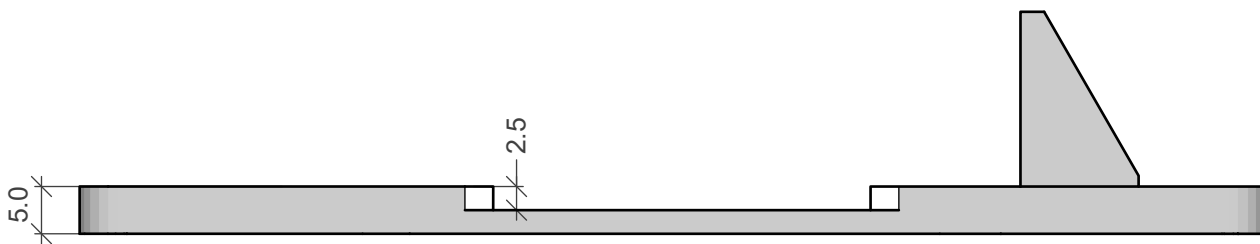
LABORATÓRIO:
LABORATÓRIO DE PROTOTIPAGEM
ASSISTIVA - LAPA

DATA:
NOV 2023

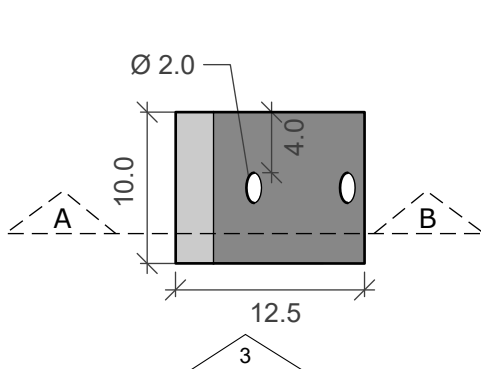
DISSERTAÇÃO DE MESTRADO

9

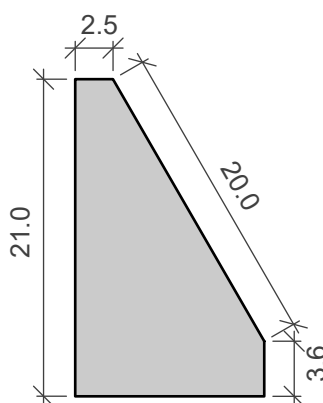
TEMA:
MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"



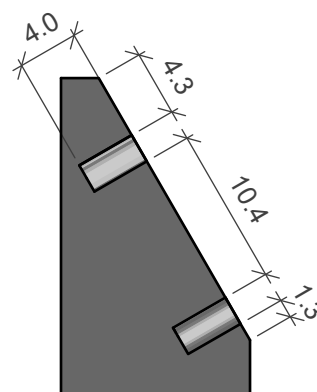
10 VISTA 2 - LATERAL DA BASE
ESCALA: 1.25:1 EM MM



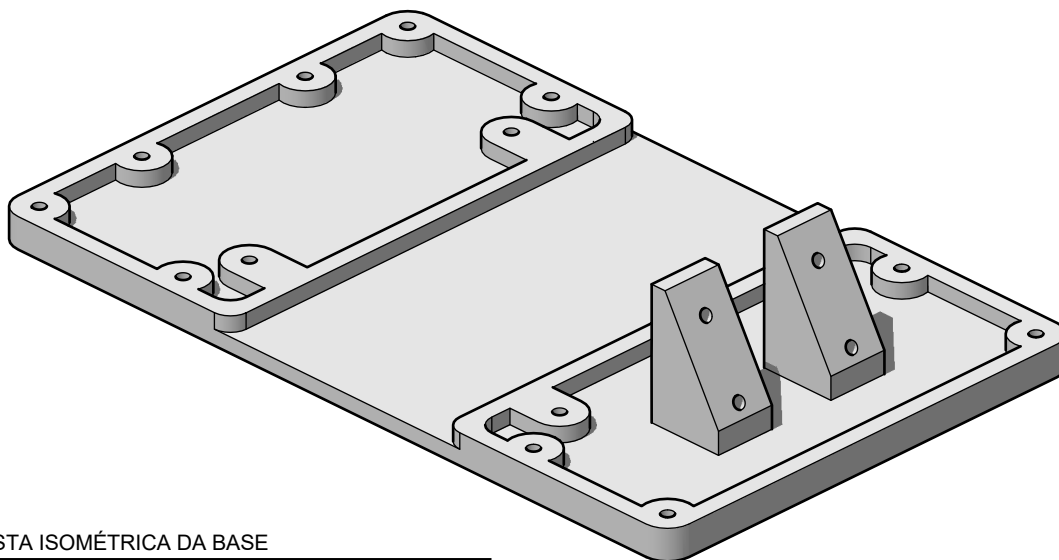
11 DETALHE 3 - APOIO DA CÂMERA
ESCALA: 2:1 EM MM



12 VISTA 3 - LATERAL DO DET 3
ESCALA: 2:1 EM MM



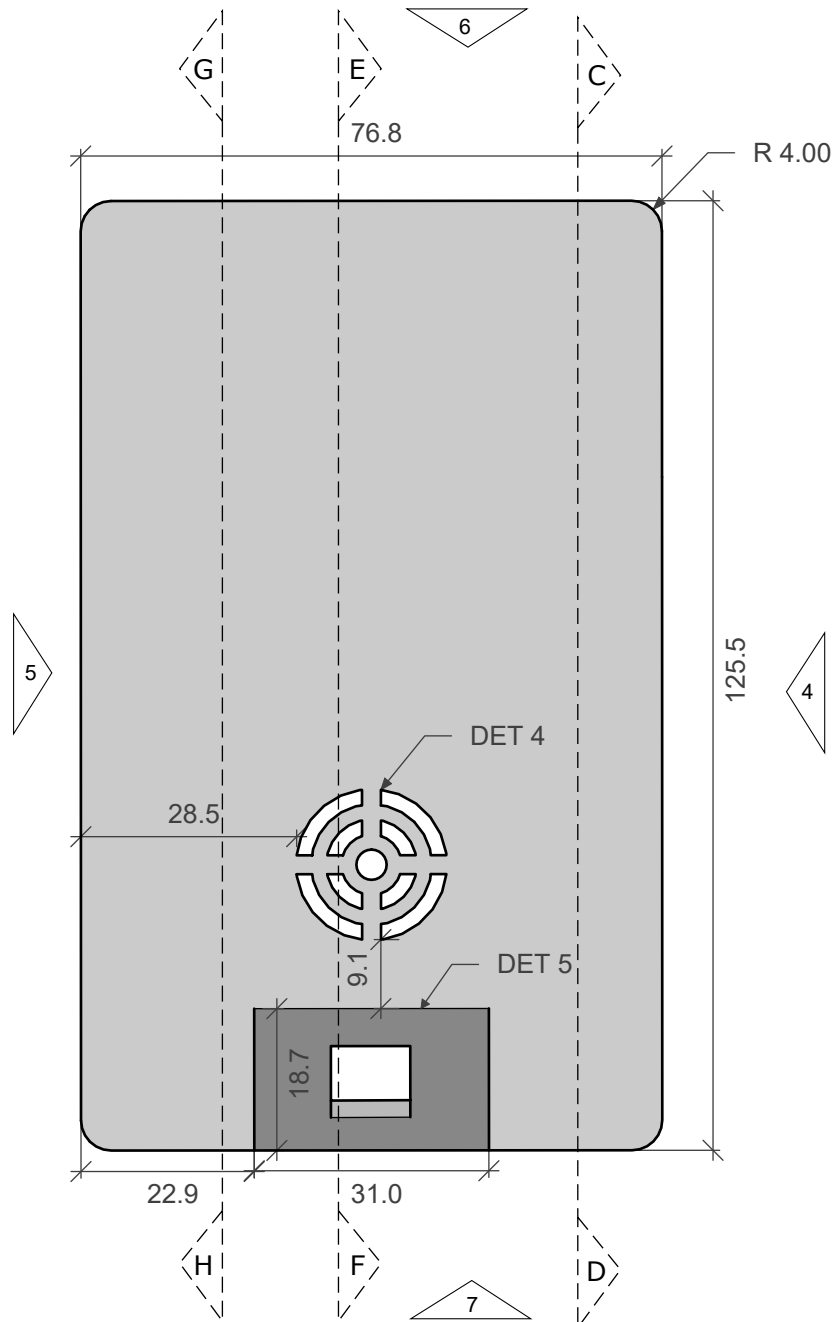
13 CORTE AB
ESCALA: 2:1 EM MM



14 VISTA ISOMÉTRICA DA BASE
ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

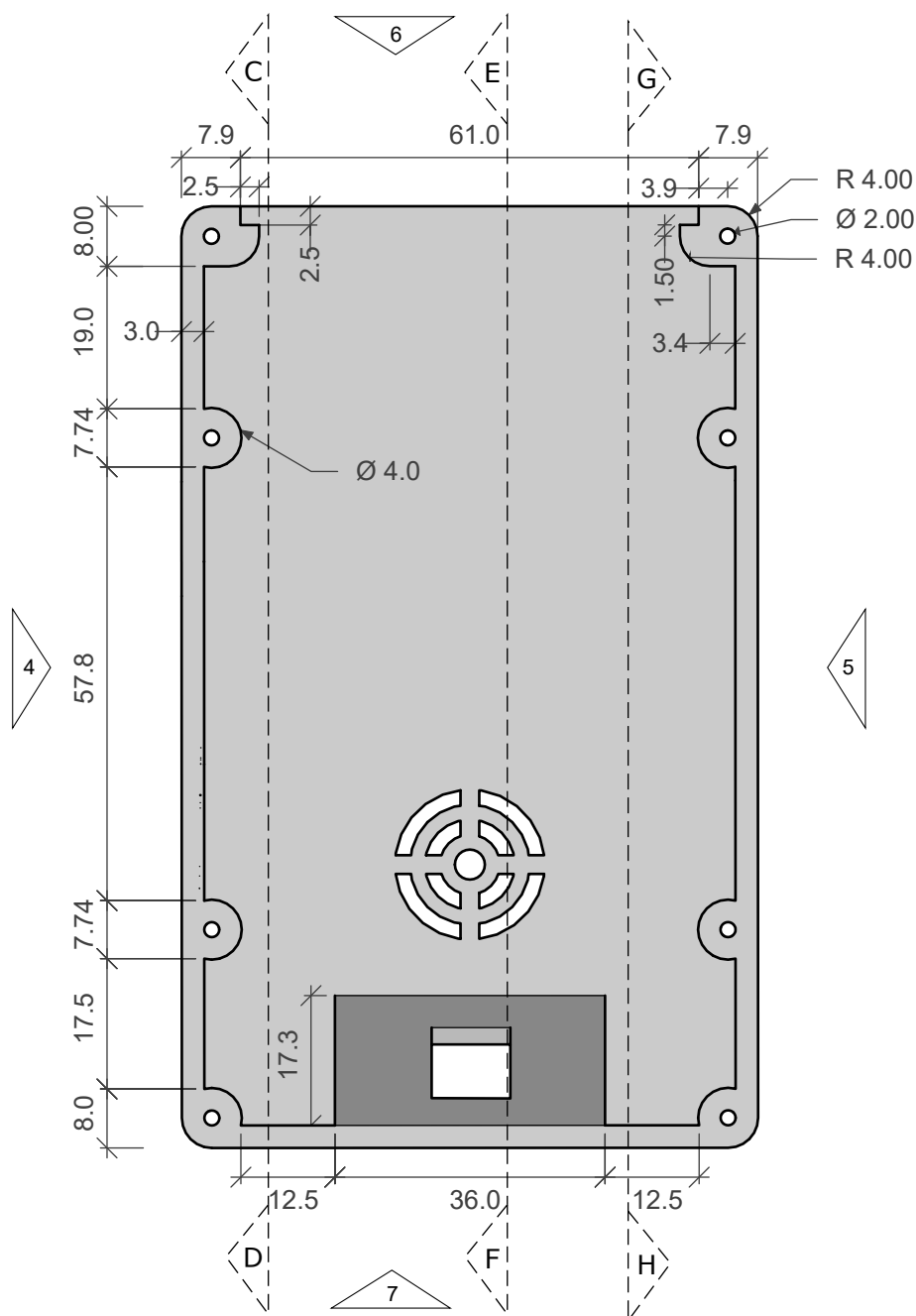
DISCENTE: LEONARDO MIRANDA ABDON		Nº DE MATRÍCULA: 202017470002	CONTEÚDO: BASE DA CAIXA PROTETORA DA PLACA RASPBERRY PI	PRANCHA Nº 3 9
ORIENTADOR: MANOEL DA SILVA FILHO		ESCALA: INDICADA	INSTITUTO DE CIÊNCIAS BIOLÓGICAS PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA CELULAR PPGNBC	
		DATA: NOV 2023	DISSERTAÇÃO DE MESTRADO	
		TEMA: MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"		
LABORATÓRIO: LABORATÓRIO DE PROTOTIPAGEM ASSISTIVA - LAPA				



15 VISTA SUPERIOR DA TAMPA
ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

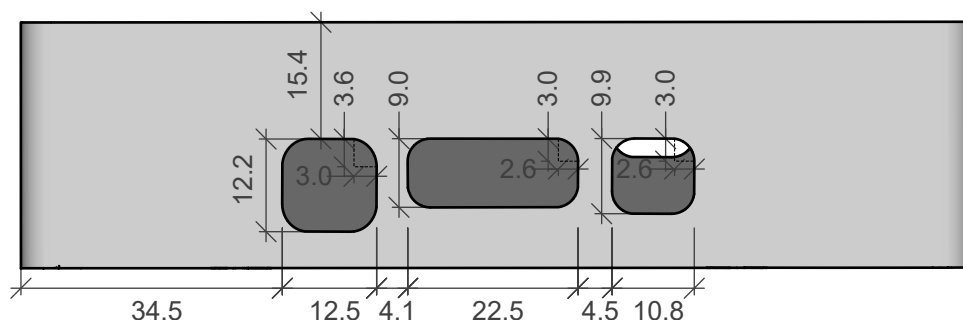
DISCENTE: LEONARDO MIRANDA ABDON		Nº DE MATRÍCULA: 202017470002	CONTEÚDO: TAMPA DA CAIXA PROTETORA DA PLACA RASPBERRY PI	PRANCHA Nº 4 9
ORIENTADOR: MANOEL DA SILVA FILHO		ESCALA: INDICADA	INSTITUTO DE CIÊNCIAS BIOLÓGICAS PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA CELULAR PPGNBC	
		DATA: NOV 2023	DISSERTAÇÃO DE MESTRADO	
		TEMA: MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"		
LABORATÓRIO: LABORATÓRIO DE PROTOTIPAGEM ASSISTIVA - LAPA				



16 VISTA INFERIOR DA TAMPA
ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

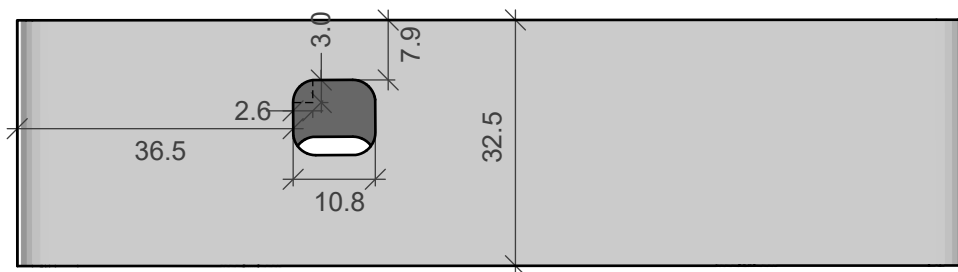
DISCENTE: LEONARDO MIRANDA ABDON		Nº DE MATRÍCULA: 202017470002	CONTEÚDO: TAMPA DA CAIXA PROTETORA DA PLACA RASPBERRY PI	PRANCHA Nº 5 9
ORIENTADOR: MANOEL DA SILVA FILHO		ESCALA: INDICADA	INSTITUTO DE CIÊNCIAS BIOLÓGICAS PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA CELULAR PPGNBC	
		DATA: NOV 2023	DISSERTAÇÃO DE MESTRADO	
		TEMA: MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"		
LABORATÓRIO: LABORATÓRIO DE PROTOTIPAGEM ASSISTIVA - LAPA				



17

VISTA 4 - LATERAL ESQUERDA DA TAMPA

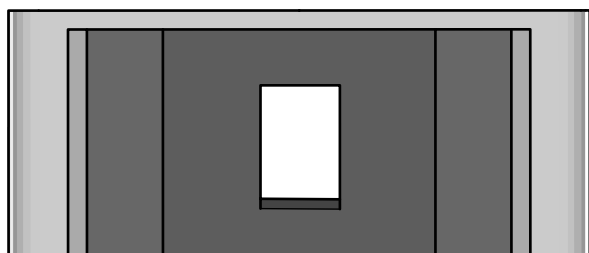
ESCALA: 1:1 EM MM



18

VISTA 5 - LATERAL DIREITA DA TAMPA

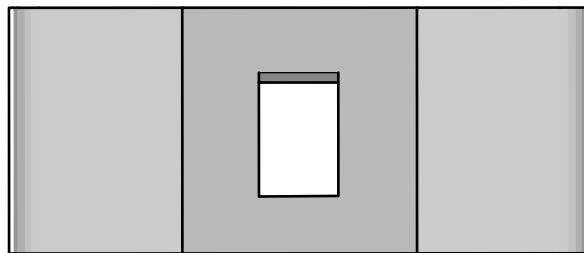
ESCALA: 1:1 EM MM



19

VISTA 6 - ABERTURA DA PORTA

ESCALA: 1:1 EM MM



20

VISTA 7 - ABERTURA DA CÂMERA

ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

DISCENTE:
LEONARDO MIRANDA ABDON

Nº DE MATRÍCULA:
202017470002

CONTEÚDO:
TAMPA DA CAIXA PROTETORA DA PLACA RASPBERRY PI

PRANCHA
Nº

ORIENTADOR:
MANOEL DA SILVA FILHO

ESCALA:
INDICADA

INSTITUTO DE CIÊNCIAS BIOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E
BIOLOGIA CELULAR PPGNBC

6

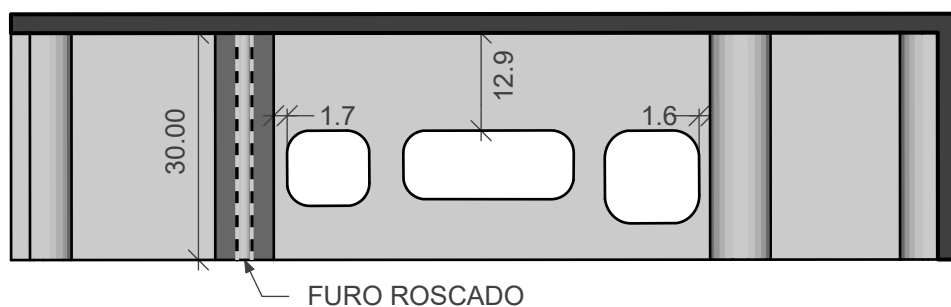
LABORATÓRIO:
LABORATÓRIO DE PROTOTIPAGEM
ASSISTIVA - LAPA

DATA:
NOV 2023

DISSERTAÇÃO DE MESTRADO

9

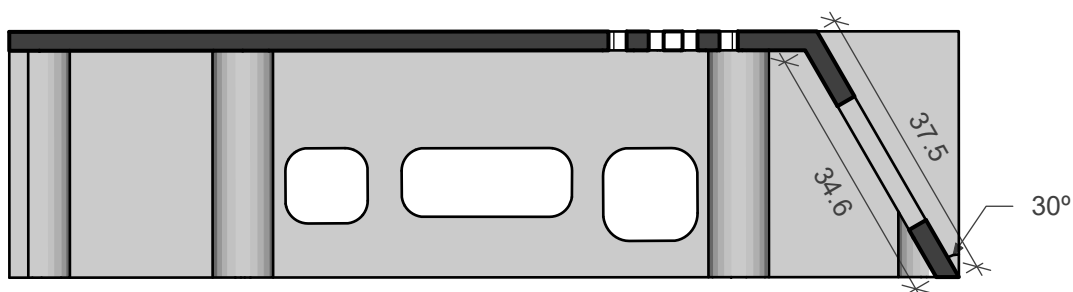
TEMA:
MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"



21

CORTE CD

ESCALA: 1:1 EM MM



22

CORTE EF

ESCALA: 1:1 EM MM



23

CORTE GH

ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

DISCENTE:
LEONARDO MIRANDA ABDON

Nº DE MATRÍCULA:
202017470002

CONTEÚDO:
TAMPA DA CAIXA PROTETORA DA PLACA RASPBERRY PI

PRANCHA
Nº

ORIENTADOR:
MANOEL DA SILVA FILHO

ESCALA:
INDICADA

INSTITUTO DE CIÊNCIAS BIOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E
BIOLOGIA CELULAR PPGNBC

7

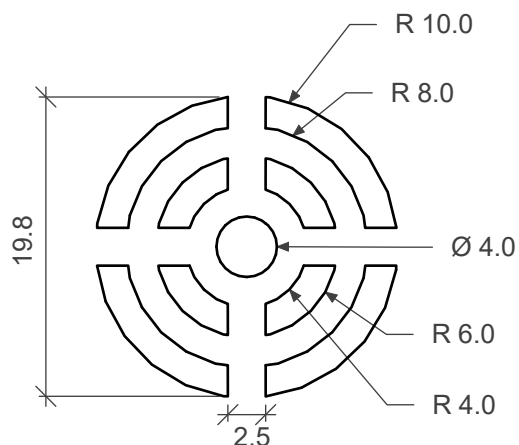
LABORATÓRIO:
LABORATÓRIO DE PROTOTIPAGEM
ASSISTIVA - LAPA

DATA:
NOV 2023

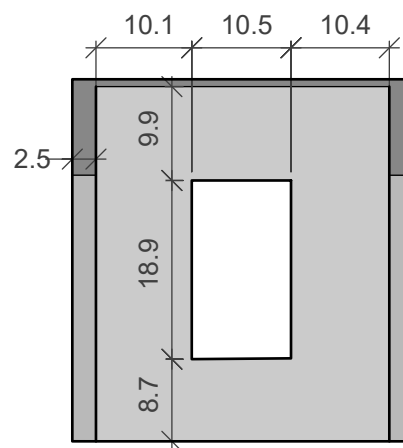
DISSERTAÇÃO DE MESTRADO

9

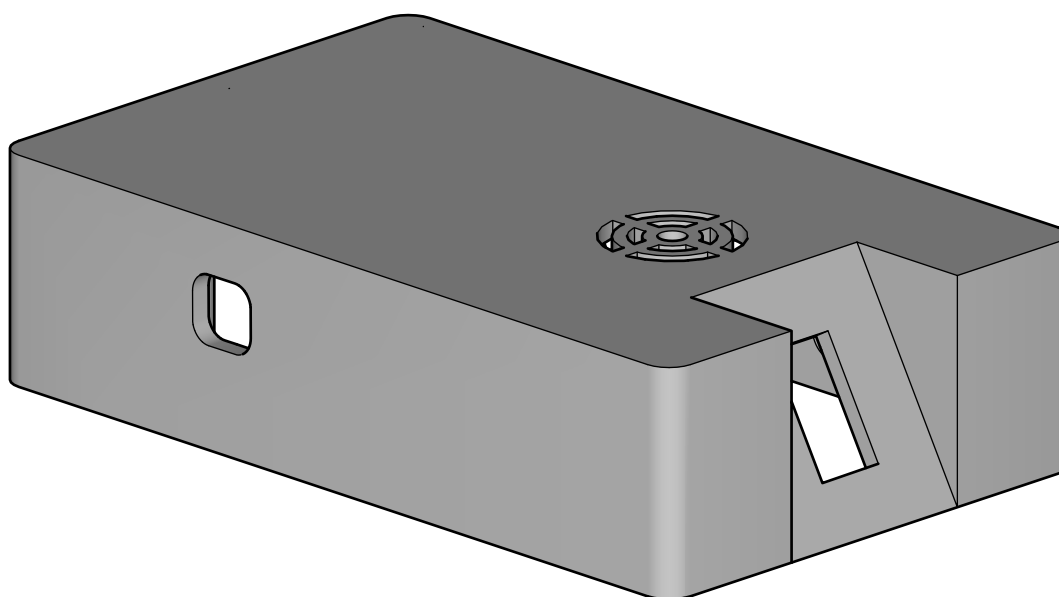
TEMA:
MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"



24 DETALHE 4 - ABERTURA DE VENTILAÇÃO
ESCALA: 2:1 EM MM



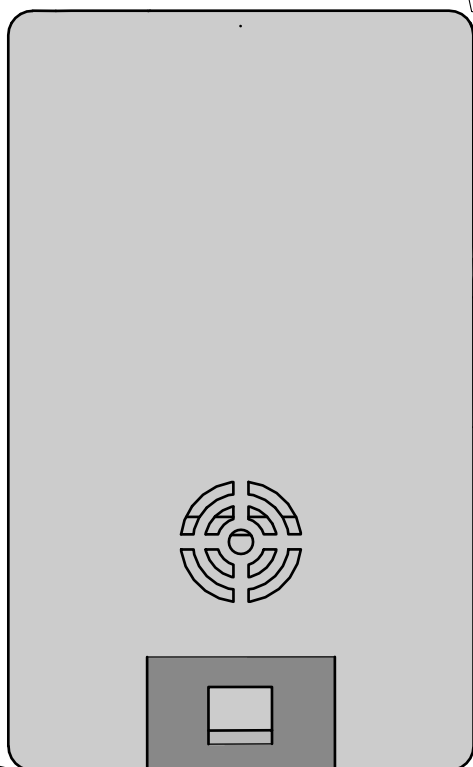
25 DETALHE 5 - ABERTURA DA CÂMERA
ESCALA: 1.25:1 EM MM



26 VISTA ISOMÉTRICA DA TAMPA
ESCALA: 1:1 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

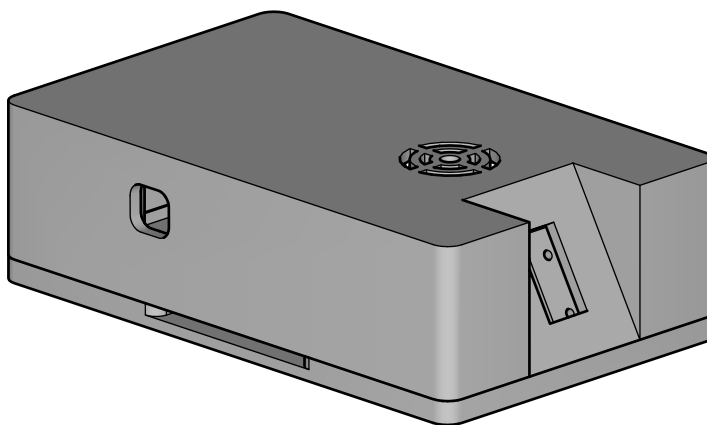
DISCENTE: LEONARDO MIRANDA ABDON	Nº DE MATRÍCULA: 202017470002	CONTEÚDO: TAMPA DA CAIXA PROTETORA DA PLACA RASPBERRY PI	PRANCHA Nº
ORIENTADOR: MANOEL DA SILVA FILHO	ESCALA: INDICADA	INSTITUTO DE CIÊNCIAS BIOLÓGICAS PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA CELULAR PPGNBC	8
LABORATÓRIO: LABORATÓRIO DE PROTOTIPAGEM ASSISTIVA - LAPA	DATA: NOV 2023	DISSERTAÇÃO DE MESTRADO	9
TEMA: MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"			



8

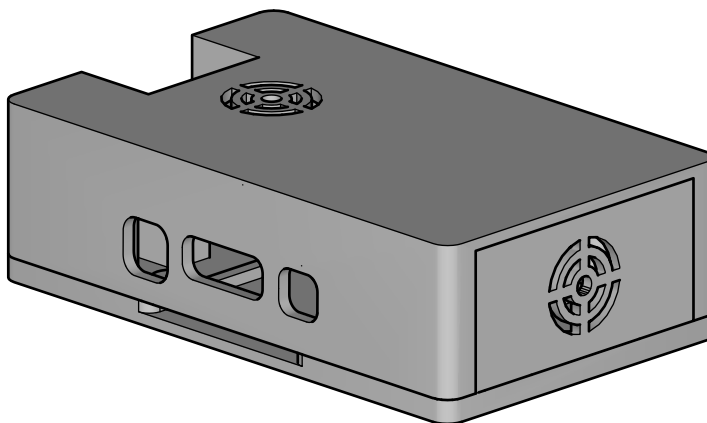
27

VISTA SUPERIOR DA CAIXA MONTADA
ESCALA: 1:1.25 EM MM



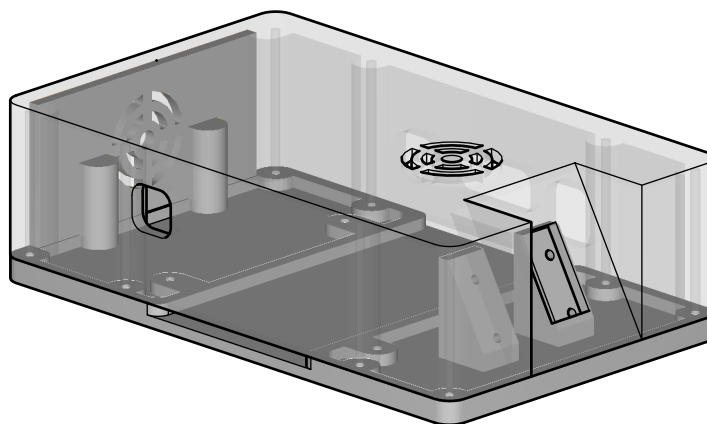
28

VISTA 8 - VISTA ISOMÉTRICA DA CAIXA MONTADA
ESCALA: 1:1.5 EM MM



29

VISTA 9 - VISTA ISOMÉTRICA DA CAIXA MONTADA
ESCALA: 1:1.5 EM MM



30

VISTA ISOMÉTRICA DA CAIXA MONTADA (TAMPA TRANSPARENTE)
ESCALA: 1:1.5 EM MM

UNIVERSIDADE FEDERAL DO PARÁ

DISCENTE: LEONARDO MIRANDA ABDON		Nº DE MATRÍCULA: 202017470002	CONTEÚDO: VISTAS ISOMÉTRICAS DA CAIXA MONTADA	PRANCHA Nº 9 9
ORIENTADOR: MANOEL DA SILVA FILHO		ESCALA: INDICADA	INSTITUTO DE CIÊNCIAS BIOLÓGICAS PROGRAMA DE PÓS-GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA CELULAR PPGNBC	
		DATA: NOV 2023	DISSERTAÇÃO DE MESTRADO	
		TEMA: MODELO DA CAIXA PROTETORA DO PROJETO "WANDERER"		
LABORATÓRIO: LABORATÓRIO DE PROTOTIPAGEM ASSISTIVA - LAPA				