O USO DE SISTEMAS EMBARCADOS ALIADOS À VISÃO COMPUTACIONAL PARA O AUXÍLIO DE DEFICIENTES VISUAIS NOS CAMPI DA UNILAB

João Matheus Lima Lúcio[1]
Paulo Roberto Gomes Abreu Filho[2]
João Gabriel Bezerra Leite[3]
Francisco Italo Guilherme Da Silva[4]
Antonio Carlos Da Silva Barros[5]

RESUMO

A significativa discrepância no baixo número de alunos com algum tipo de deficiência visual no ensino superior é o cenário utilizado para esse projeto. A partir dele, vem a importância de se reconhecer as dificuldades que os portadores de deficiência visual enfrentam no seu dia-a-dia. A criação de um dispositivo que permita alertar, através de sinais sonoros, obstáculos e objetos identificados à frente do usuário, é uma solução para locomoção de deficientes visuais dentro dos campi da universidade que poderá permitir uma melhor inclusão deste aluno na universidade. Este trabalho tem como objetivo desenvolver um sistema de baixo custo para facilitar a acessibilidade e inserção das pessoas deste grupo na sociedade e, principalmente, no ensino superior. O dispositivo proposto neste trabalho é composto de um serviço de visão computacional desenvolvido através dos frameworks YOLO e Tensorflow Lite para identificação de objetos, além de serviços para suporte ao usuário, tradução de texto-para-fala e detecção de queda. Os modelos embarcados no dispositivo são capazes de detectar diversos objetos. Como forma de validação, foram realizados testes para medir o desempenho desse conjunto. Os serviços obtiveram um desempenho melhor, e mais próximo do esperado. Em geral, o dispositivo mostrou ser possível ser solução para o problema que se propõe a solucionar. Entretanto, devido a pandemia não foi possível realizar a homologação ou validação por um deficiente visual por conta da não disponibilidade de voluntários. Além disso, também não foi possível realizar os testes do equipamento desenvolvido nos campi da universidade.

Palavras-chave: visão computacional deficiente visual acessibilidade serviços .

Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Instituto de Engenharia e Desenvolvimento Sustentavel, Discente, joaomatheuslimalucio@aluno.unilab.edu.br[1]
Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Instituto de Engenharia e Desenvolvimento Sustentável, Discente, paulo@aluno.unilab.edu.br[2]
Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Instituto de Engenharia e Desenvolvimento Sustentável, Discente, bljoaogabriel@aluno.unilab.edu.br[3]
Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Instituto de Engenharia e Desenvolvimento Sustentável, Discente, italoguilherme18.1@aluno.unilab.edu.br[4]
Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Instituto de Engenharia e Desenvolvimento Sustentavel, Docente, carlosbarros@unilab.edu.br[5]

ISSN: 2447-6161

INTRODUÇÃO

Dada a constante evolução da tecnologia humana, problemas tão antigos quanto à própria humanidade recebem um novo olhar e, consequentemente, novas soluções. A dificuldade de inclusão e acessibilidade a uma melhor condição de vida para pessoas com algum tipo de deficiência, que significa uma interação de impedimentos de longo prazo seja de natureza física, mental, intelectual ou sensorial, na participação em sociedade em condição igualdade com as demais pessoas, de acordo com o Art. 2º da Lei Brasileira de Inclusão (LBI), 13.146 de julho de 2015, é justamente o problema a ser tratado nesse exposto. Segundo o Instituto Brasileiro de Geografia e Estatística, IBGE, 45,61 milhões de pessoas no Brasil sofrem algum tipo de deficiência. Desses, 18 por cento possuem deficiência visual. (IBGE, 2010). Ainda segundo o IBGE, existem cerca de 8,45 milhões de estudantes no ensino superior, e desses, só 43.633 são deficientes. (SALLIT, 2018). Já, sendo mais específico, o número de deficientes visuais em um cenário mais atual no ensino superior é de apenas 29.221. (BARROS, SANTANA e MARTINS, 2019). É a partir dessa problemática que surge a necessidade de uma solução que permita, mesmo que de maneira simples, a melhoria na qualidade de vida do deficiente visual, e, por consequência, aumento no número desses no ensino superior.

O sistema E-BLIND se propõe a ser para a Unilab (Universidade da Integração Internacional da Lusofonia Afro-Brasileira) esta solução. Um dispositivo com um sistema do tipo embarcado, que significa um sistema computacional especializado que faz parte de uma máquina ou sistema maior (BEAL, 1997, tradução nossa), que visa auxiliar o deficiente visual na navegação por entre os campi da universidade. Ele vem equipado com uma placa Raspberry Pi, que é definido por FIORIN e ANTONELLO (2017, p. 2) como: um minicomputador do tamanho de um cartão de crédito. E esse que ainda possui vantagens como baixos consumo de energia e preço, tamanho reduzido, ser silencioso e móvel. Sua função é capturar imagens e realizar a detecção de objetos através de uma tecnologia chamada visão computacional, definida por "[...] é a área da ciência que estuda e procura desenvolver teorias e métodos voltados à extração automática de informações úteis contidas em imagens." (CROWLEY E CHRISTENSEN, 1997 apud SILVA, 2008).

METODOLOGIA

Para o desenvolvimento deste dispositivo seguiu-se as seguintes etapas: Definição de arquitetura, Treinamento do modelo, Engenharia do protótipo e Montagem e teste do protótipo.

Abaixo será descrito cada etapa do desenvolvimento do Sistema E-BLIND.

ETAPA 1: DEFINIÇÃO DA ARQUITETURA.

A arquitetura do dispositivo é representada na Figura 1.

Figura 1: arquitetura do Sistema E-BLIND.

Fonte: Elaborado pelo autor.

Na Figura 1 é possível identificar os seguintes elementos da arquitetura:

ISSN: 2447-6161

Raspberry PI: Responsável pela aquisição das imagens, coletar dados dos sensores e realizar a detecção de objetos. A placa utilizada na arquitetura é a Raspberry Pi 3 Modelo B Rev 1.2.
Sensor Ultrasonico: Responsável em calcular a distância dos objetos em relação ao usuário. O modelo utilizado foi o sensor ultrassônico da Adafruit HC-SR04.
Acelerômetro: Sensor utilizado para detecção de movimentos abruptos classificados como uma queda do usuário. O modelo utilizado foi o sensor acelerômetro de três eixos ADXL345 da Adafruit.
Câmera: O sensor de câmera é proprietário, chamado de módulo câmera, e é comercializado pela Raspberry Pi. O modelo utilizado foi o OmniVision OV5647 (V1).

ETAPA 2: TREINAMENTO DO MODELO

O primeiro passo é definir quais e quantos objetos se deseja detectar, chamados de classes. O segundo é montar um conjunto de imagens, chamado muitas vezes de dataset, para o treinamento e validação do modelo. O treinamento compreende a parte de extração de características específicas aos objetos, classes, e usar essas características para a montagem do modelo. O terceiro é usar parte do dataset, previamente separada, para realizar testes no modelo. Utilizando o framework YOLO/Darknet, o treinamento foi feito com um dataset compreendido da seguinte maneira:

Classes: Backpack (Mochila), Chair (Cadeira), Door (Porta), Stairs (Escada), Car (Carro), Bus (Ônibus) e Person (Pessoa).
Quantidade total de imagens: 107890
Quantidade total de caixas delimitadoras: 107890
Tamanho total do dataset: 16,3 gigabytes

Após o treinamento dos pesos, nome dado ao modelo, esse é, então, convertido para o framework Tensorflow Lite, usado especificamente em dispositivos móveis com um desempenho razoável mesmo sem uma placa gráfica dedicada.

ETAPA 3: ENGENHARIA DO PROTÓTIPO

A engenharia, a montagem e o teste do protótipo podem ser definidos como passos interligados. A fim de fazer o dispositivo funcionar da melhor forma possível, foram definidos os seguintes componentes: o dispositivo de captura e o dispositivo de processamento de dados. Como seu objetivo é capturar as cenas à frente do usuário e descrever a posições de elementos detectados, posição, nome e distância, as configurações podem ser descritas como: dispositivo de captura de dados: faz a coleta de dados como a imagem da câmera e dos sensores de proximidade e aceleração; e dispositivo de processamento de dados: compreendido pela placa Raspberry Pi, sua fonte de energia, cabos dos sensores e câmera e do fone de ouvido, faz o processamento dos dados os transformando em informação. Os hardwares são, sim, interligados via cabo no protótipo criado. A mobilidade não é tão prejudicada por esse fato, já que esse ainda funciona.

ETAPA 4: MONTAGEM E TESTE DO PROTÓTIPO

Nesta etapa será montado o protótipo para testes e validação do mesmo. A montagem inicial será realizada em protoboard e a montagem final em caixa com toda arquitetura embarcada.

ISSN: 2447-6161

RESULTADOS E DISCUSSÃO

O protótipo final desenvolvido neste trabalho é apresentado na Figura 2, sendo composto por um dispositivo de captura de dados e do dispositivo de processamento de dados (Raspberry Pi).

Figura 2 — Montagem do Protótipo da arquitetura final.

Fonte: Elaborado pelo autor.

Na etapa de treinamento foi possível obter os modelos detalhados abaixo:

Modelo	mAP (mean Average Precision)
23000	47.74
Melhor	43.21
Último	43.51

A tabela demonstra as métricas atingidas pelos modelos após o treinamento. O primeiro, 23000, tem seu nome pela sua iteração, que são passos durante o treino em que o modelo termina com o dataset e faz testes. Como visto na Tabela 3 o Modelo 23000 é o mais preciso. O modelo de nome “melhor” é selecionado pelo framework como melhor durante uma determinada quantidade de iterações. Tem performance pior que o primeiro pois é de um ciclo diferente. O último, assim como seu nome, foi o último modelo gerado durante o treinamento. Seu desempenho ruim possui uma explicação através de um fenômeno bem comum nesse campo. Conhecido como overfitting, acontece quando um modelo aprende detalhes e barulho sobre os dados para treinamento a ponto de impactar negativamente a performance do modelo sobre novos dados. Isso significa que o barulho ou flutuações aleatórias nos dados de treino são extraídos e aprendidos pelo modelo. O problema é que esses conceitos não se aplicam a novos dados e impactam negativamente na habilidade do modelo de generalizar. (BROWNLEE, 2016, tradução nossa).

Apesar de todo o trabalho e os meses de treinamento usando a ferramenta de uso de GPU da Google, Google Colab, o modelo final foi mudado devido a uma falha na conversão para o framework Tensorflow. Para a prototipagem, então, um modelo pré-treinado e desenhado para dispositivos móveis, com baixa latência, foi escolhido direto da página, chamada de zoológico, da TensorFlow, o SSD MobileNet V2 FPNLite 640x640. Sua acurácia é definida como: mAP 29.1 e identifica até 90 objetos ou classes.

Com o modelo embarcado foram realizados testes de autonomia e percebeu-se que a bateria de 20000mA obteve uma duração estimada de 16h. Os sensores se mostraram sem muita inferência, resultando em uma boa acurácia. A placa Raspberry encontrou um pouco de dificuldade ao rodar o modelo Tensorflow e lidar com o processamento dos outros sensores, o que resultou em uma baixa contagem de quadros por segundo. O protótipo registrou uma temperatura máxima de 70º celsius, algo abaixo do limite da placa, resultando em algo considerado razoável.

O software embarcado foi feito de maneira a processar os dados capturados e transformar esses dados em informação a partir de 4 serviços desenvolvidos na linguagem Python. Os serviços são:

ISSN: 2447-6161

Serviço de Fala, que conta com um código em python utiliza a biblioteca nativa de tradução de texto-para-fala;
Serviço de Detecção de Objetos, que utiliza um código escrito em Python que consome o modelo pré-treinado disponibilizado pelo framework TensorFlow com 90 classes de objetos;

Serviço de Medição de Distância de Obstáculos, que utiliza um sensor ultrassônico para medir a distância de um obstáculo e avisar sobre uma possível colisão, além de controlar a detecção de objetos e só permitir detecções a partir de 4 metros do objeto; e

Serviço de Detecção de Queda, que utiliza um sensor acelerômetro para analisar o movimento e manda um código com a localização do usuário para o sistema de suporte em caso de queda.

CONCLUSÕES

O trabalho desenvolveu um protótipo capaz de detectar objetos e informar sobre obstáculos. O dispositivo contemplou a interação do usuário com um cenário com possibilidades para detecção de obstáculos com suas localizações indicadas como objeto a direita e objeto a esquerda. Através dos algoritmos de texto para voz o sistema informa ao usuário os objetos detectados e obstáculos encontrados. Infelizmente não foi possível validar e aperfeiçoar o protótipo para obter uma melhor usabilidade e utilização por parte dos deficientes visuais nos campi da universidade devido a pandemia covid-19. Acredita-se que o dispositivo trará impactos positivos para os alunos beneficiados tanto no aprendizado técnico como na sua visão profissional de contribuir com a sociedade dispondo-se de seus conhecimentos adquiridos na graduação. Por fim, espera-se que a ação maior desenvolvida nas turmas apresente um modelo para as gestões municipais, com fatores positivos no desenvolvimento regional, nacional e dos países parceiros.

AGRADECIMENTOS

À Universidade da Integração Internacional da Lusofonia Afro-Brasileira (UNILAB) pela oportunidade de desenvolvimento do projeto e sua apresentação, à Pró-Reitoria de Extensão pelo, Arte e Cultura (Proex) pelo apoio financeiro e ao orientador, professor Dr. Antonio Carlos da Silva Barros, pelo aprendizado e por sua disponibilidade.

REFERÊNCIAS

BRASIL. Lei nº 13.146, de 6 de julho de 2015. Institui a Lei Brasileira de Inclusão da Pessoa com Deficiência. Diário Oficial da União: seção 1, Brasília, DF, ano CLII, n. 127, p. 2, 7 julho. 2015.
IBGE. Censo Demográfico 2010: Características gerais da população, religião e pessoas com deficiência. Disponível em: . Acesso em: 10 de mai. 2021
SALLIT, Mathias. As maiores representatividades de pessoas com deficiência nas universidades do Brasil. Revista Quero, 2 de dez. de 2018. Disponível em: . Acesso em: 10 de mai. de 2021.
BARROS, A.; SANTANA, A.; MARTINS, R. Inclusão: relatos de deficientes visuais no ensino superior brasileiro. Livro-Reportagem em Revista, abr./jun. de 2019. Disponível em: . Acesso em: 10 de mai. de 2021.

ISSN: 2447-6161

BEAL, Vangie. Embedded System. Webopedia, 15 de agos. de 1997. Disponível em: . Acesso em: 10 de mai. de 2021.

FIORIN, P.; CARVALHO, H.; ANTONELLO, R. Domótica: Desenvolvimento de protótipo para automação residencial com smartphone android e raspberry pi. SECTEC, IV. Santa Catarina, 2017.

SILVA, G. Desenvolvimento de um Sistema de Visão Computacional para o Estudo do Comportamento de Animais Experimentais. Tese (Mestrado em Engenharia Elétrica) – Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais. Belo Horizonte, p. 98. 2008.

BROWNLEE, Jason. Overfitting and Underfitting With Machine Learning Algorithms. Machine Learning Mastery, 21 de mar. de 2016. Disponível em: . Acesso em: 10 de mai. de 2021.

ISSN: 2447-6161