http://journal.unoeste.br/index.php/ce/index
DOI: 10.5747/ce.2021.v13.n2.e359
ISSN on-line 2178-8332

Colloquium Exactarum

Submetido: 10/03/2021 Revisado: 27/05/2021 Aceito:01/08/2021

MAKEMESEE – UMA SOLUÇÃO PARA O AUXÍLIO DE DEFICIENTES VISUAIS

MakeMeSee – An aid to help visual impairment people

Henrique Ramos Ricci[1], Francisco Assis da Silva[1], Mário Augusto Pazoti[1]

[1]Universidade do Oeste Paulista
E-mail: ricci.henrique@outlook.com; chico@unoeste.br; mario@unoeste.br

RESUMO – Soluções envolvendo inteligência artificial e visão computacional têm se tornado cada vez mais comuns nos últimos anos, devido ao aumento do poder computacional e o desenvolvimento de novas tecnologias. Essas soluções abrangem boa parte das necessidades humanas, como carros autônomos, segmentação de imagens médicas ou previsões para o mercado financeiro. Visto que a acessibilidade também é uma área muito importante e que as técnicas de inteligência artificial e visão computacional podem proporcionar soluções que auxiliem pessoas com deficiência, neste trabalho é abordada uma solução que permite a detecção, cálculo e narração de obstáculos para auxiliar portadores de deficiência visual. Por meio de um hardware composto por duas webcams, capaz de fazer a captura de imagens diferentes de uma mesma cena, e de um *software* capaz de processar as imagens obtidas, classificando e detectando os obstáculos, a solução visa informar ao usuário o que está a sua frente.
Palavras-chave: deep learning; visão computacional; deficiência visual.

ABSTRACT – Solutions involving artificial intelligence has become increasingly common in the last years because the increase of computer power and emergence of new technologies. These works include many humans needs, like autonomous cars, segmentation of medical images or financial market predictions. Since accessibility is a very important area and the techniques like artificial intelligence and computer vision can make solutions to help disabled people, this paper is showing an aid that can detect, calculate and narrate obstacles to help visual impairment people. With a hardware compound by two webcams, responsible by to get different images from a scene, and a software that can processing the images, classifying, and detecting the obstacles, the system can inform the user what are ahead.
Keywords: deep learning; computer vision; visual impairment.

1. INTRODUÇÃO

Segundo a Organização Mundial da Saúde (OMS, 2018), estima-se que a nível mundial, aproximadamente 1,3 bilhões de pessoas vivem com alguma forma de distúrbio visual, das quais cerca de 441 milhões possuem um grau entre moderado e cegueira total. No Brasil, segundo o Instituto Brasileiro de Geografia e Estatística e a fundação Dorina, cerca de 1,2 milhões de

brasileiros sofrem com algum destes graus mais altos (IBGE, 2010) (DORINA, 2012).

Para o Ministério da Educação, a deficiência visual (DV) é a perda ou redução da capacidade visual em ambos os olhos em caráter definitivo, em que não há melhora ou correção mesmo com procedimentos clínicos/cirúrgicos ou uso lentes corretoras (MEC, 2010).

Uma das maiores complicações de pessoas portadoras de DV é a dificuldade em perceber do mundo, tendo que se adaptar a entendê-lo e senti-lo por meio de outros sentidos. Outro problema, é a realização de tarefas em locais que a pessoa desconheça, por se tratar de um ambiente totalmente diferente do que ela está acostumada.

Atualmente, diversas formas de inclusão para pessoas que possuem deficiência já foram elaboradas e colocadas em prática como as famosas bengalas, piso tátil nas calçadas e os cães-guias, por exemplo. Todas as soluções permitem um certo grau de liberdade para os portadores da deficiência caminharem em ambientes abertos, mas não proporcionam uma sensação semelhante a real, de como é o local e, no caso do piso tátil, nem sempre está disponível em todas as calçadas.

Além disso, segundo o instituto IRIS (2002), o custo para se treinar um cão guia é de aproximadamente R$ 35.000,00. Também é importante ressaltar que o tempo de espera pode chegar a três anos e nem sempre há cães disponíveis, uma vez que treiná-los é uma tarefa extremamente complicada e demorada. Devido ao tempo de espera, essa solução pode acabar se distanciando um pouco da realidade das pessoas.

Uma alternativa promissora é uso da computação como ferramenta para a inclusão de diversos tipos de deficiência e com relação aos problemas visuais não é diferente. Autores como Zhang e Caspi (2019), Wang *et al*. (2016), Wang *et al.* (2017), Sousa (2013), Shi *et al.* (2017), Sepehrinour e Kasaei (2017), Poggi e Mattoci (2017), Pinheiro Lima Neto *et al.* (2017), Karkar, Puthren e Al-Maadeed (2018), Kacorri *et al.* (2017), Gianani *et al.* (2018), Elmannai e Elleithy (2018), Chowdhury, Gao e Islam (2016) e Castro (2018) trazem em seus trabalhos algumas soluções, que vão desde *softwares* que auxiliam no uso de computadores, a ferramentas que tentam trazer as sensações do mundo para os usuários.

Para facilitar a utilização do computador, Costa e Barros (2006) propuseram uma ferramenta composta por um bloco de notas e uma calculadora, de forma que através de retornos por voz, o deficiente pode saber o que está digitando, enviar email para outros usuários, salvar e imprimir textos e ainda fazer operações matemáticas básicas, que normalmente geram complicações por necessitarem de visualização.

Diversos outros *softwares* proporcionam uma melhor interação entre usuário e computador, narrando conteúdos em tela com síntese de voz ou braile, mas ainda assim, as interações se tornam restritas a um ambiente interno.

Uma das técnicas computacionais que podem auxiliar no processo de inclusão de portadores de DV é a Visão Computacional (VC). Segundo a empresa DeepAI (2019), visão computacional é um campo de estudo da ciência da computação que gira em torno de como os computadores podem ver e entender imagens digitais e vídeos. Por meio de um processamento dos dados extraídos das imagens, pode-se obter algumas informações relevantes sobre a cena, como pessoas contidas nela. Com a intenção de ampliar o alcance das técnicas de VC, algoritmos de Inteligência Artificial (IA) foram incorporadas ao longo dos últimos anos, trazendo resultados ainda mais precisos.

Visto esses problemas, nota-se a necessidade de utilizar alguma tecnologia que permita amenizar tais dificuldades. Como exposto, existem soluções na área de computação que resolvem alguns desses problemas, porém são soluções que muitas vezes dependem da conexão com um computador remoto, possuem um hardware muito grande ou mesmo não são precisas (restringindo o uso em ambientes externos e se tornando desconfortável para o usuário). Também foi exposto que algoritmos de *Deep Learning* são muito eficazes para as melhorias na VC, porém essa técnica é pouco usada na área de DV, impedindo o portador de utilizar aplicações que poderiam ajudá-lo de alguma forma, mostrando que há uma necessidade de realizar estudos nessas áreas.

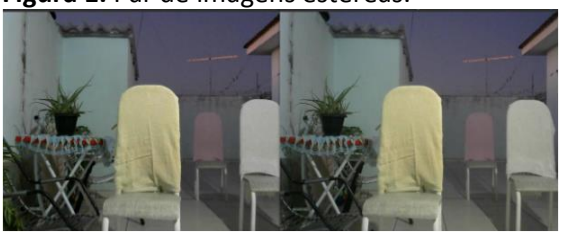
2. FUNDAMENTAÇÃO TEÓRICA

Nesta seção é apresentada a fundamentação teórica, em que são expostos os conceitos e técnicas para o desenvolvimento do resultado deste trabalho.

2.1. Estereoscopia

A estereoscopia está associada à visão binocular (dois olhos). Segundo Tommaselli (2009), visão binocular é um dos processos que permitem a percepção de profundidade de uma cena. Cada olho enxerga uma mesma cena de pontos de vista diferentes. A junção dessas imagens faz com que o cérebro crie uma cena 3D e assim, consiga a noção de profundidade. No âmbito da computação, essa técnica pode ser implementada por meio de duas câmeras posicionadas de forma estéreo. As imagens obtidas por cada uma das câmeras (Figura. 1) são corrigidas, processadas e então é possível gerar um mapa de profundidade, que após mapeado posteriormente, permite saber as distâncias dos objetos contidos na cena.

Figura 1. Par de imagens estéreas.

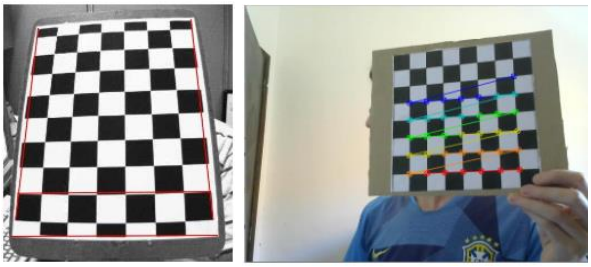


Fonte: Os autores.

2.2. Calibração de Câmera

Câmeras digitais nunca são fabricadas da mesma maneira. Em toda a fabricação de lentes sempre ocorrem algumas variações que impactam no resultado da imagem, causando leves distorções nas capturas. Uma das formas de realizar a correção dessas distorções é utilizando uma técnica chamada calibração de câmera. Por meio de algum objeto que possui as distâncias iguais entre alguns pontos (neste trabalho foi utilizado um tabuleiro de xadrez) é encontrada a matriz de parâmetros intrínsecos da câmera, que contém as informações de distância focal e centros óticos. Essas informações permitem uma recolocação dos pontos obtidos originalmente para uma nova projeção sem a distorção causada pelas lentes. Na Figura 2 pode-se observar a distorção da imagem e o processo de correção.

Figura 2. Em (a) é possível ver as distorções das bordas do tabuleiro. Em (b) é exemplificado o mapeamento do tabuleiro para a calibração.



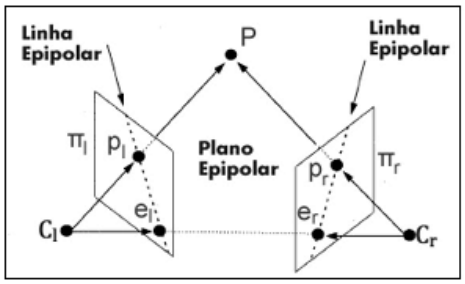
(a) (b)

Fonte: Os autores.

2.3. Mapa de Disparidade

Após terem sido obtidas duas imagens estéreas sem distorções, é possível gerar um mapa de disparidade. Esse mapa contém as informações da diferença de distância entre cada pixel da imagem da esquerda e da direita. Para melhorar a eficiência da geração do mapa, são utilizados conceitos de geometria epipolar. Ao invés de buscar ponto a ponto de uma imagem a outra, é feita a busca de pelo menos oito pontos correlatos. Por meio desses pontos são geradas duas matrizes, essencial e fundamental. Elas simplificam a busca de novos pontos correlatos, na qual ao invés de buscar os pontos na imagem inteira, é necessário buscá-los somente na linha epipolar (Figura 3). Esse processo diminui o tempo de processamento, tornando o algoritmo mais eficiente.

Figura 3. Exemplificação da geometria epipolar.



Fonte: (Bessa *et al.*, 2016).

Tendo sido gerado o mapa de disparidade, uma associação entre cor do mapa e distância real do objeto pode ser realizada para a geração do mapa de profundidade, que informa a qual distância um determinado objeto da cena se encontra. Uma observação importante é que o mapa é convertido para a escala de cinza para

facilitar a visualização e o mapeamento. Na Figura 4 é mostrado o resultado do mapa de disparidade em comparação com a imagem da câmera esquerda.

Figura 4. Geração do mapa de disparidade.



Fonte: Os autores.

2.4. Deep Learning

As redes neurais com arquiteturas *Deep Learning* se espelham no modelo de neurônio artificial convencional (ROSENBLATT, 1958) com a diferença de que podem existir muitos neurônios nas camadas ocultas assim como podem existir muitas camadas ocultas na rede. Essas múltiplas camadas ocultas permitem uma melhor separação dos dados, o que torna a técnica perfeita para projetos que envolvem a classificação de muitas classes.

O funcionamento da arquitetura *Deep Learning* é o mesmo do *Perceptron*: encontrar o melhor peso para cada neurônio a fim de maximizar o resultado da classificação. Isso se dá por meio da repetição constante dos dados de treinamento, funções de ativação e o algoritmo *backpropagation*, que permite a correção dos pesos utilizando como base erros obtidos. A representação de um neurônio artificial utilizado na arquitetura *Deep Learning* pode ser representada pela Equação 1, onde xj representa a entrada, wkj representa o peso referente à rede e uk representa o valor de saída do neurônio.

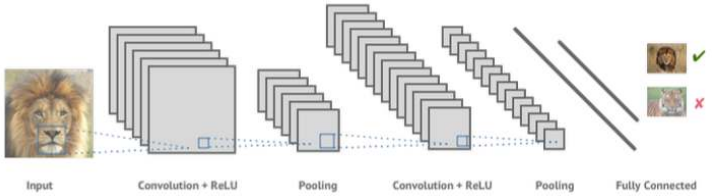
$$u_k = \sum_{j=1}^{m} 1\, x_j\, w_{kj} \qquad (1)$$

2.5. Redes Neurais Convolusionais

Utilizar imagens originais para treinamento de redes neurais pode ser muito custoso, uma vez que cada *pixel* da imagem é uma entrada para a rede. Para minimizar essas entradas, geralmente são utilizadas redes neurais convolucionais (*Convolutional Neural Network* – CNN).

As CNNs consistem em uma rede neural multicamadas que no início do processo, aplicam diversas máscaras de convolução na imagem para extrair algumas de suas principais características. Em seguida, é aplicada uma técnica chamada *Pooling*, que extrai apenas as informações mais relevantes da convolução gerada. Isso permite a redução no número de entradas para a rede neural profunda e ainda filtra dados que são irrelevantes para o processo. Esse processo de convolução e *Pooling* ocorre quantas vezes forem necessárias. Ao final, as matrizes geradas são transformadas em uma estrutura unidimensional e enviadas à rede neural para classificação. A Figura 5 mostra o processo completo de uma rede neural convolucional.

Figura 5. Processo de uma rede neural convolucional. Dada a imagem do leão, por exemplo, as camadas de convolução e *Pooling* extraem as melhores características e enviam para a rede neural profunda realizar a classificação.



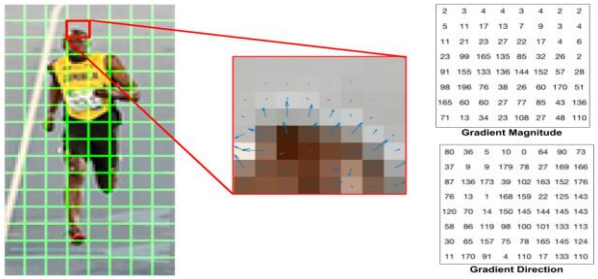
Fonte: (ELECTRICAL LIBRARY, 2018).

2.6. HOG (*Histogram of Oriented Gradient*)

O HOG é um descritor que analisa a forma e a textura de objetos. Seu funcionamento se resume na obtenção de duas matrizes chamadas de matriz de direção e matriz de intensidade. A matriz de intensidade armazena as informações da maior variação de cor entre os *pixels* vizinhos. A matriz de direção armazena a informação referente ao ângulo de variação do pixel com a mais intensa.

A Figura 6 mostra as matrizes de intensidade e direção, juntamente com um exemplo gráfico para auxiliar no entendimento.

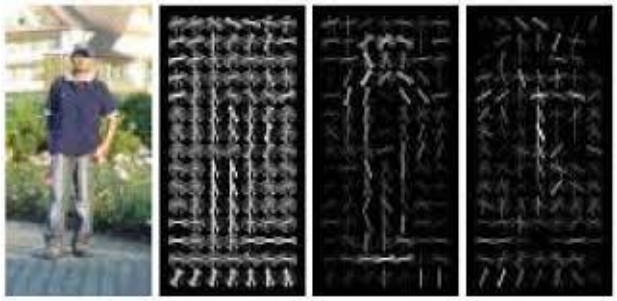
Figura 6. Extração das matrizes de intensidade e direção.



Fonte: (DALAL e TRIGGS, 2015).

Obtidas as matrizes, cria-se o histograma com base na análise dos valores, no qual o objetivo é criar o contorno do objeto analisado por meio dos histogramas e vetores (Figura 7).

Figura 7. Exemplo dos vetores gerados pelo HOG.



Fonte: (PAJDLA et al, 2009).

2.7. KNN (*K-Nearest Neighborhood*)

O KNN é um algoritmo da classe de inteligência artificial utilizado para classificação. Seja N um conjunto de dados rotulados, K a distância dos vizinhos e X uma nova entrada. Por meio do cálculo da distância entre todos os dados de N, o algoritmo classifica X como sendo pertencente a moda das classes entre as K menores distâncias. Geralmente o cálculo de distância mais utilizado esse algoritmo é a distância euclidiana, mas distâncias como *Manhattan* ou *Minkowski* também podem ser utilizadas.

3. METODOLOGIA

O trabalho foi estruturado em três partes para facilitar o desenvolvimento e os testes (montagem das câmeras, cálculo de distâncias, reconhecimento facial). A primeira parte planejada foi a estruturação das duas câmeras no suporte, de forma que essa montagem é capaz de capturar duas imagens da mesma cena, porém de ângulos diferentes. O suporte é mostrado na Figura 8. Esse suporte foi conectado a um dispositivo *Android* por meio de um adaptador USB. Essas imagens são processadas em seguida, a fim de detectar os obstáculos no caminho e calcular a distância em relação ao usuário.

Figura 8. Hardware desenvolvido.

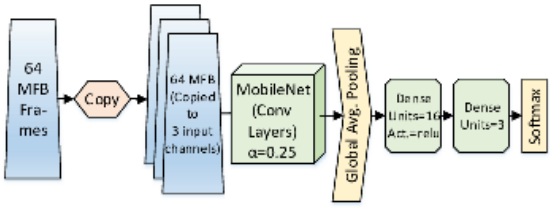


Fonte: Os autores.

A integração com o *smartphone Android* foi feita por meio de uma técnica chamada JNI (*Java Native Interface*) que permite a inclusão de códigos C++ em aplicativos desenvolvidos em *Java*. Com isso, é possível acessar informações de baixo nível dos dispositivos, além de proporcionar uma maior velocidade de processamento. Essa técnica foi utilizada tanto para a integração das webcams, quanto para o processamento do reconhecimento facial, que será abordado posteriormente.

Após a integração, foi necessário a criação dos algoritmos para detecção e reconhecimento dos obstáculos. Para isso, foram utilizadas as imagens obtidas pela câmera posicionada a esquerda do usuário. A cada passo dado pelo usuário, um frame é capturado e enviado para uma CNN baseada em uma arquitetura chamada SSD (*Single Shot Detection*) *Mobilenet*, muito eficaz para classificações em dispositivos móveis. Essa rede foi treinada por HOWARD *et al.* (2017) e sua arquitetura é mostrada na Figura 9. Essas imagens são submetidas a rede, treinada com o *dataset* Ms. Coco, capaz de classificar até 80 classes distintas, sendo as mais comuns encontradas neste trabalho: carros, motos, pessoas, cães, placas de trânsito e objetos encontrados em casas, como televisores e geladeiras. Ao ser realizada a detecção, são calculadas as distâncias de cada objeto e, caso estejam a menos de 8 metros do usuário, são então alertados por meio de áudio.

Figura 9. Arquitetura *MobileNet*.



Fonte: (HUSSIAN M. S. e HAQUE, M., 2018).

Antes de realizar o cálculo das distâncias, é necessário corrigir as distorções das câmeras, para não gerar resultados imprecisos. Para isso, foi realizado a calibração das câmeras em um computador externo. As informações de correções (matriz de parâmetros intrínsecos e distâncias focais) foram salvas em arquivos JSON e, ao iniciar a aplicação, é realizada a leitura e configuração dos parâmetros dentro do algoritmo. Isso permite que cada frame obtido pelas câmeras seja corrigido e então realizados os cálculos.

Para o cálculo de distância dos obstáculos, são utilizados os conceitos de estereoscopia, geometria epipolar e geração do mapa de disparidade. Após a detecção das imagens pela câmera esquerda, é realizada a junção com a imagem obtida pela câmera direita, realizando o mapeamento de pontos correlatos nas duas imagens para a geração do mapa de disparidade. Por meio da geometria epipolar, não é necessário encontrar todos os pontos correlatos de uma vez (aumentando a eficiência do algoritmo). Ao invés disso, é realizado a busca de apenas 16 pontos, e então gerado a matriz essencial e fundamental, que permitem a busca de novos pontos de maneira mais eficiente. Na Figura 10 é mostrado a execução do sistema, sendo a imagem de cima a câmera direita, a imagem do meio a câmera esquerda e na parte de baixo a detecção dos obstáculos e o mapa de disparidade.

Figura 10. Sistema em execução.



Fonte: Os autores.

Por fim, foi realizada a implementação de algoritmos para o treinamento e reconhecimento facial. A fim de obter um treinamento mais rápido, foi utilizado uma técnica de extração de características faciais onde, para cada imagem de rosto submetida ao algoritmo, um vetor contendo as informações da face é gerado, sendo salvo para a aplicação do algoritmo de classificação, que neste trabalho foi o KNN. Ao identificar um novo rosto, é feita a extração das características faciais aplica-se o KNN para descobrir a qual classe esse novo rosto pertence.

4. RESULTADOS

Os resultados obtidos foram satisfatórios para a proposta (precisão acima de 75% de precisão e tempo de execução menor que 100 ms, meta essa definida no início do projeto), uma vez que tanto o tempo de execução quanto a precisão atingiram resultados viáveis para o uso real da solução.

Com relação ao tempo de execução, o reconhecimento dos obstáculos leva cerca de 60ms para o reconhecimento de todos os obstáculos da cena. Variações na iluminação e quantidade de obstáculos podem aumentar ou diminuir esse tempo, mas nunca passando de 90 ms, sendo extremamente rápido e garantindo o reconhecimento antes do usuário colidir. Em relação ao reconhecimento facial, esse é o algoritmo mais lento do sistema, levando cerca de 700 ms para realizar uma detecção. Inicialmente esses valores eram de cerca de 3000 ms, sendo necessário a aplicação de ROI (*Region of Interest* – Região de Interesse) para otimizar esse tempo. Nesse tempo descrito também se encontram inclusos os tempos de cálculo de distâncias, que individualmente levam cerca de 13 ms.

Com relação à precisão dos cálculos, o algoritmo se mostrou muito eficaz, tendo como margem de erro aproximadamente 30 cm. Isso se da por conta da função obtida, que relaciona cor do mapa de disparidade com a distância real do obstáculo. Durante a fase de verificação, foram posicionados obstáculos a 25 cm de distância uns dos outros, em uma faixa que variou de 2 a 11 metros em relação às câmeras. Todas as distâncias e cores foram mapeadas, gerando a função que tem como entrada a cor e gera como saída a distância. Para identificar qual cor pertence a qual obstáculo, foi utilizado a *bouding box* fornecida pela rede neural, que retorna à posição em que o objeto está. Tendo essa informação, é gerado um histograma das cores que se encontram dentro dessa região e então é filtrado somente as cores mais relevantes. Feito isso, uma média desses valores é obtida e esse resultado é enviado para a função.

Já com relação à precisão do reconhecimento de obstáculos, os resultados foram de cerca de 84 % de forma estáticas, ou seja, detecções feita com o usuário parado em algum lugar. Em movimento esse valor diminui para cerca de 72 %, devido ao arrasto das imagens gerado pelas *webcams*, uma vez que elas não foram feitas para serem utilizadas em movimento. Esse arrasto também impacta na geração do mapa de disparidade, podendo comprometer o resultado. Isso pode ser facilmente resolvido utilizando câmeras adequadas para capturas de imagens em movimento, como câmeras de ação (GoPro Hero ou Sony RX0 Mark II, por exemplo), porém seu custo é mais elevado.

Não foram realizados testes com relação à precisão de desvio dos obstáculos e nem testes com relação a precisão do reconhecimento facial, uma vez que a base de rostos era muito pequena e os resultados poderiam ser tendenciosos.

5. CONSIDERAÇÕES FINAIS

Visto todo o exposto, nota-se que o trabalho atingiu resultados satisfatórios para a proposta sugerida, permitindo que a aplicação possa ser utilizada em ambientes reais com alguns poucos ajustes.

Existem muitas melhorias que serão feitas em projetos futuros, principalmente em relação a precisão dos cálculos e reconhecimentos e de integrações que auxiliem ainda mais o usuário, como narração de rotas e detecção de trânsito, que permitirão que o indivíduo possa realmente se locomover com conforto e segurança.

Também devem ser feitas algumas alterações nos algoritmos implementados, principalmente com relação ao reconhecimento facial, visando aprimorar a precisão do reconhecimento e diminuindo o tempo de processamento.

REFERÊNCIAS

BESSA, J.; BESSA, R.; XAVIER, A**. Sistema de determinação de coordenadas cartesianas baseado em visão estéreo**. 2016. Disponível em: https://www.researchgate.net/publication/304217514_SISTEMA_DE_DETERMINACAO_DE_COORDENADAS_CARTESIANAS_BASEADO_EM_VISAO_ESTEREO. Acesso em: 14 dez. 2020.

CASTRO, M. D. **Alvisku :** uso da visão computacional e sons 3D para auxílio a cegos. 2018.

CHOWDHURY, M.; GAO, J.; ISLAM, R. Distance Measurement of Objects using Stereo Vision. HELLENIC CONFERENCE ON ARTIFICIAL INTELLIGENCE. 9., **Proceedings** [...]. SETN ’16.Thessaloniki, Greece: Association for Computing Machinery, 18 maio 2016. Disponível em: https://doi.org/10.1145/2903220.2903247. Acesso em: 26 mar. 2020. https://doi.org/10.1145/2903220.2903247

CONDE, A. J. M. **Definição de cegueira e baixa visão.** 2016. Disponível em: http://www.ibc.gov.br/images/conteudo/AREAS_ESPECIAIS/CEGUEIRA_E_BAIXA_VISAO/ARTIGOS/

Def-de-cegueira-e-baixa-viso.pdf. Acesso em: 02 set. 2019.

COSTA, E. B. G; BARROS, I. O. R. Matraca – Ferramenta Computacional Para Auxílio A Deficientes visuais no uso do computador. *In*: ENAPET, 11., 2006, Florianópolis. **Anais** [...]. Florianópolis, 2006. Disponível em: http://www.enapet.ufsc.br/anais/MATRACA_FERRAMENTA_COMPUTACIONAL_PARA_AUXILIO_A_DEFICIENTES_VISUAIS_NO_USO_DO_COMPUTADOR.pdf. Acesso em: 05 set. 2019.

DALAL, N. TRIGGS, B. **Histograms of oriented gradients for human detection**. 2005. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, v. 1, p. 886-893.

DEEPAI. **What is Computer Vision?** 2019. Disponível em: https://deepai.org/machine-learning-glossary-and-terms/computer-vision. Acesso em: 01 set. 2019.

DEEPAI. **What is Deep Learning?** 2019. Disponível em: https://deepai.org/machine-learning-glossary-and-terms/deep-learning. Acesso em: 01 set. 2019.

DORINA - FUNDAÇÃO DORINA NOWILL PARA CEGOS. **Estatísticas da deficiência visual**. Disponível em: https://www.fundacaodorina.org.br/a-fundacao/deficiencia-visual/estatisticas-da-deficiencia-visual/. Acesso em: 25 ago. 2019.

ELECTRIAL E-LIBRARY. **O que são redes neurais convolucionais?** 2018. Disponível em: https://www.electricalelibrary.com/2018/11/20/o-que-sao-redes-neurais-convolucionais/ . Acesso em: 14 dez. 2020.

ELMANNAI, W. M.; ELLEITHY, K. M. A **Highly Accurate and Reliable Data Fusion Framework for Guiding the Visually Impaired**. IEEE Access, v. 6, p. 33029–33054, 2018. https://doi.org/10.1109/ACCESS.2018.2817164

GHILARDI, M. C. *et al.* Real-Time Detection of Pedestrian Traffic Lights for Visually Impaired People. 2018 International Joint Conference on Neural Networks (IJCNN). Anais [...] *In*: 2018 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN). jul. 2018. https://doi.org/10.1109/IJCNN.2018.8489516

GIANANI, S. et al. **JUVO - An Aid for the Visually Impaired.** 2018 International Conference on Smart City and Emerging Technology (ICSCET). Anais... *In*: 2018 INTERNATIONAL CONFERENCE ON SMART CITY AND EMERGING TECHNOLOGY (ICSCET). jan. 2018. https://doi.org/10.1109/ICSCET.2018.8537270

HOWARD, A. G.; ZHU, M.; CHEN, B.; KALENICHENKO, D.; WANG, W.; WEYAND, T.; ANDREETTO, M.; ADAM, H. **MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.** 2017. Disponível em: https://arxiv.org/abs/1704.04861. Acesso em: 05 Jul. 2021.

HUSSIAN M. S. e HAQUE, M. **SwishNet: A Fast Convolutional Neural Network for Speech, Music and Noise Classification and Segmentation.** 2018. Disponível em: https://www.researchgate.net/publication/328654938_SwishNet_A_Fast_Convolutional_Neural_Network_for_Speech_Music_and_Noise_Classification_and_Segmentation. Acesso em: 05 Jul. 2021.

IBGE. **Características gerais da população, religião e pessoas com deficiência**. 2010. Disponível em: https://censo2010.ibge.gov.br/apps/mapa/. Acesso em: 04 set. 2019.

IRIS. **Instituto IRIS cão-guia**. Disponível em: http://www.iris.org.br/. Acesso em: 05 set. 2019.

KACORRI, H. et al. **People with Visual Impairment Training Personal Object Recognizers: Feasibility and Challenges**. Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. Anais...: CHI '17.Denver, Colorado, USA: Association for Computing Machinery, 2 maio 2017. Disponível em: https://doi.org/10.1145/3025453.3025899. Acesso em: 26 mar. 2020. https://doi.org/10.1145/3025453.3025899

KARKAR, A.; PUTHREN, M.; AL-MAADEED, S. **A Bilingual Scene-to-Speech Mobile Based Application**. 2018 International Conference on Computer and Applications (ICCA). Anais... In: 2018 INTERNATIONAL CONFERENCE ON

COMPUTER AND APPLICATIONS (ICCA). ago. 2018. https://doi.org/10.1109/COMAPP.2018.8460239

MEC. **Alunos com Deficiência visual: Baixa visão e cegueira**. 2010. Disponível em: http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=7105-fasciculo-3-pdf&Itemid=30192. Acesso em: 04 set. 2019.

PAJDLA, T. et al. **Incongruence detection for detecting, removing, and repairing incorrect functionality in low-level processing**. 2009. Disponível em: https://www.researchgate.net/publication/228982025_Incongruence_detection_for_detecting_removing_and_repairing_incorrect_functionality_in_low-level_processing. Acesso em: 14 dez 2020.

PINHEIRO LIMA NETO, E. *et al.* **Sensory Substitution of Vision: A Systematic Mapping and a Deep Learning Object Detection Proposition**. 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI). Anais... In: 2019 IEEE 31ST INTERNATIONAL CONFERENCE ON TOOLS WITH ARTIFICIAL INTELLIGENCE (ICTAI). nov. 2019. https://doi.org/10.1109/ICTAI.2019.00274

POGGI, M.; MATTOCCIA, S. **A wearable mobility aid for the visually impaired based on embedded 3D vision and deep learning**. 2016 IEEE Symposium on Computers and Communication (ISCC). Anais... In: 2016 IEEE SYMPOSIUM ON COMPUTERS AND COMMUNICATION (ISCC). jun. 2016. Disponível em: <https://ieeexplore.ieee.org/document/7543741>. https://doi.org/10.1109/ISCC.2016.7543741

ROSENBLATT, F. **The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain**. 1958. Psychological Review, 65—386. https://doi.org/10.1037/h0042519

SEPEHRINOUR, M.; KASAEI, S. **Perspective reconstruction of non-rigid surfaces from single-view videos**. 2017 Iranian Conference on Electrical Engineering (ICEE). Anais... In: 2017 IRANIAN CONFERENCE ON ELECTRICAL ENGINEERING (ICEE). maio 2017. https://doi.org/10.1109/IranianCEE.2017.7985271

SHI, Q.; MA, L.; YANG, Y**. Practical Method of Low-Light-Level Binocular Ranging Based on Triangulation and Error Correction.** Proceedings of the 2017 International Conference on Computer Science and Artificial Intelligence. Anais...: CSAI 2017.Jakarta, Indonesia: Association for Computing Machinery, 5 dez. 2017. Disponível em: <https://doi.org/10.1145/3168390.3168417>. Acesso em: 26 mar. 2020. https://doi.org/10.1145/3168390.3168417

SOUSA, K. A. O. **Uso de visão computacional em dispositivos móveis para auxílio à travessia de pedestres com deficiência visual**. 2013. 85 f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Presbiteriana MACKENZIE, São Paulo, 2013. Disponível em: http://tede.mackenzie.br/jspui/handle/tede/1432. Acesso em: 05 set. 2019.

TOMMASELLI, A. M. G. **Fotogrametria Básica - Introdução**. 2009. p. 144-175. Disponível em: http://www.faed.udesc.br/arquivos/id_submenu/891/introducao_a_fotogrametria.pdf. Acesso em 05 mai 2020.

WANG, H. S. et al. **Enabling independent navigation for visually impaired people through a wearable vision-based feedback system**. 2017. IEEE International Conference on Robotics and Automation, 2017. Disponível em: https://ieeexplore.ieee.org/abstract/document/7989772. Acesso em 05 set. 2019. https://doi.org/10.1109/ICRA.2017.7989772

WANG, Y.-X. et al. **ThirdEye: a coaxial feature tracking system for stereoscopic video see-through augmented reality**. ACM SIGGRAPH 2016 Posters. Anais...: SIGGRAPH '16.Anaheim, California: Association for Computing Machinery, 24 jul. 2016. Disponível em: <https://doi.org/10.1145/2945078.2945100>. Acesso em: 26 mar. 2020. https://doi.org/10.1145/2945078.2945100

ZHANG, Y.; CASPI, A. **Stereo Imagery Based Depth Sensing in Diverse Outdoor Environments: Practical Considerations**. Proceedings of the 2nd ACM/EIGSCC Symposium on Smart Cities and Communities. Anais...: SCC '19.Portland, OR, USA: Association for Computing Machinery, 10 set. 2019. Disponível em: <https://doi.org/10.1145/3357492.3358627>.

Acesso em: 26 mar. 2020. https://doi.org/10.1145/3357492.3358627