

Aplicação do método YOLO para o reconhecimento de caracteres Braille

Roberto P. do Nascimento
Universidade Federal do Rio
Grande do Sul (UFRGS)
Porto Alegre – RS – Brasil
robertotpd@gmail.com

Lorenzo A. F. Aragão
Universidade Federal do Oeste
do Pará (UFOPA)
Santarém – PA – Brasil
lorenzoalberto02@gmail.com

Josivan R. Reis
Universidade Federal do Oeste
do Pará (UFOPA)
Oriximiná – PA – Brasil
josivanrreis@gmail.com

Regina O. Heidrich
Universidade FEEVALE
Novo Hamburgo – RS – Brasil
rheidrich@feevale.br

Dante A. C. Barone
Universidade Federal do Rio
Grande do Sul (UFRGS)
Porto Alegre – RS – Brasil
barone@inf.ufrgs.br

ABSTRACT

The Braille system is characterized as an essential resource in the education of students with visual impairment (VI). However, the lack of knowledge of Braille spelling on the part of the teacher has led to difficulties in written communication, which depends exclusively on specialists, affecting educational activities. Thus, to overcome the difficulties of communication written in Braille between DVs and sighted people, this work presents a model of automatic recognition of Braille with its correspondent in Portuguese, using You Only Look Once (YOLO). The developed method is robust and capable of identifying Braille characters with an accuracy above 97%. In the future, the functional performance of the method will be tested through an accessible user interface.

RESUMO

O sistema Braille se caracteriza como um recurso essencial na formação dos alunos com deficiência visual (DV). No entanto, o desconhecimento da grafia Braille por parte do professor tem gerado dificuldade na comunicação escrita, dependendo exclusivamente de especialistas, o que prejudica as atividades educativas. Dessa forma, para superar as dificuldades de comunicação escrita em Braille entre DVs e videntes, este trabalho apresenta um modelo de reconhecimento automático do Braille com a letra correspondente em português, por meio do método You Only Look Once (YOLO). O método projetado é robusto e capaz de identificar os caracteres Braille com precisão acima de 97%. No futuro, o desempenho funcional do método será testado por meio de interface de usuário acessível.

Author Keywords

Tradução Braille; YOLO; Professor Vidente.

CCS Concepts

•Applied computing → Computer-assisted instruction;

INTRODUÇÃO

A comunicação inclusiva cria condições para que a sociedade desperte para o fato de que, se tratando de pessoas, o que importa é sua capacidade de realização, evolução, produção, convivência, o potencial de criação, e não suas limitações. A educação inclusiva, por sua vez, não discrimina o aluno em função de qualquer desigualdade ou diferença no acesso à educação e aprendizagem, garantindo assim, a participação de qualquer aluno no processo de ensino e de aprendizagem nas salas de aula regular [6]. Mas para que a inclusão no âmbito educacional ocorra de forma satisfatória, os professores videntes¹ das escolas de ensino regular, deverão estar aptos para atender a comunicação escrita dos alunos com DV.

Em [3] é mencionado que a comunicação bidirecional assimétrica, como mostra a Figura 1, descreve um desequilíbrio na comunicação entre indivíduos, levando frequentemente na perda ou alteração das informações. Ainda segundo os autores, há menos métodos de comunicação que permita videntes entender Braille e, como isso, é necessário desenvolver tecnologias que ajudem a equilibrar essa assimetria na comunicação para promover o fluxo de informações e a comunicação entre videntes e DV.

Com base na Lei Brasileira de Inclusão da Pessoa com Deficiência², no Brasil, todas as crianças que estão em idade escolar, sem exceção, devem frequentar as salas de aulas do ensino regular. Contudo, a maioria dos educadores que tem em suas

¹Termo usado para descrever pessoas que não têm limitações visuais graves

²https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm



Figure 1. Ilustração da comunicação assimétrica entre videntes e DV. Adaptado de [3]

turmas alunos com DV, não é proficiente em Braille. Isso impacta no processo de ensino e de aprendizagem, pois o Braille é frequentemente usado para comunicação entre pessoas cegas e videntes. Em particular, a situação é bastante comum quando professores videntes trabalham com alunos cegos e eles têm que lidar com livros didáticos e trabalhos de alunos escritos em Braille [7].

Assim, uma das estratégias comumente adotadas, nas escolas e nas universidades, para superar esse problema, é enviar o trabalho do aluno com DV para um profissional qualificado no sistema Braille para realizar a tradução do texto para português e depois enviado de volta ao professor da sala de aula regular para, só então, corrigir a atividade. Isso torna o processo oneroso, tanto para os professores, quanto para os alunos [13].

Por outro lado, os professores de Educação Especial que atuam como mediadores, traduzindo os textos e elaborando materiais pedagógicos acessíveis para alunos com DV, embora sejam especialistas em Educação Especial não são em outras áreas do ensino, como, por exemplo, no ensino básico, português, ciências entre outros [14].

Essa dificuldade tem gerado uma grande lacuna na comunicação escrita entre professores videntes e alunos com DV, também entre os seus pares, ou seja, entre os alunos vidente e os alunos com DV, gerando dificuldades no processo de ensino e de aprendizagem. Com isso, este trabalho tem como objetivo construir um método capaz de reconhecer Braille, a partir de imagens Braille, apresentando o caractere correspondente em português. O estudo busca auxiliar os professores videntes, envolvidos na educação de crianças com DV, no reconhecimento do sistema Braille, de modo a melhorar a comunicação e tornar mais fácil e rápido fornecer feedback ao aluno sem a necessidade de um intermediário, principalmente, no quesito de leitura. Além disso, colegas e responsáveis legais dos estudantes com DV poderão utilizar o sistema para se comunicar com as crianças.

SISTEMA BRAILLE

Criado na França, em 1825, por Louis Braille, o sistema de leitura para deficientes visuais foi inspirado na técnica de pontos e traços usada por soldados para a comunicação durante a noite. O sistema Braille comumente aceito consiste em 64 caracteres (incluindo o vazio como um sinal), onde cada cela consiste em duas colunas de três pontos, cada uma disposta de

forma retangular em forma de matriz 3x2, como exibido na Figura 2.

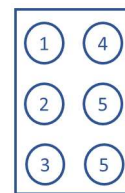


Figure 2. Célula Braille

Os recentes avanços na tecnologia têm oferecido às pessoas com DV novas oportunidades para receber e transmitir a informação, pois a leitura e a escrita em Braille continuam sendo um importante meio de comunicação para eles [7]. No entanto, essas novas tecnologias têm levado as pessoas a perguntarem se ainda é importante aprender Braille. É importante reforçar que o Braille agrega outros valores ao DV, não sendo apenas um meio de comunicação escrita, e sim, abrindo caminho para alfabetização, conhecimento, informação, liberdade intelectual, igualdade de oportunidades, segurança pessoal, dentre outros. É por isso que o DV deve aprender Braille, mesmo com as opções tecnológicas à disposição [10].

Como o Braille é um alfabeto feito de pontos gravados em papel, como pode ser visto na Figura 3, o texto é representado visualmente por pontos em relevo brancos em um fundo branco, os professores videntes que tem algum conhecimento em Braille, geralmente, leem os textos com os olhos, no entanto, seu reconhecimento visual é extremamente desafiador, principalmente quando há texto impresso no verso do papel [7]. A utilização de meios técnicos para reconhecimento em Braille, também pode facilitar muito este trabalho, auxiliando os videntes na leitura e compreensão dos documentos Braille.

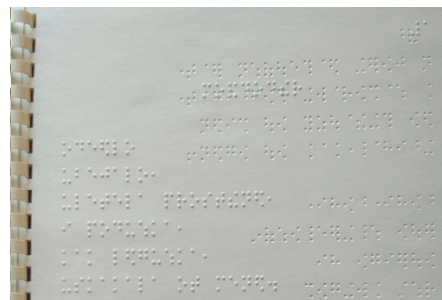


Figure 3. Exemplo de um Manuscrito Braille [4]

MATERIAIS E MÉTODOS

A abordagem proposta possui 5 etapas. A etapa 1 é usada para aquisição dos caracteres Braille do *dataset* público da plataforma Kaggle [9]. Até onde foi possível pesquisar, este *dataset* é o único disponível publicamente com caracteres Braille rotulado, de grau 1, ou seja, que não é contraído, o que significa que há um caractere Braille específico para cada letra, totalizando 1.560 imagens das 26 letras do alfabeto, de "a" a "z", com imagem de 28x28 pixels em escala de cinza, como pode ser observado na Figura 4.

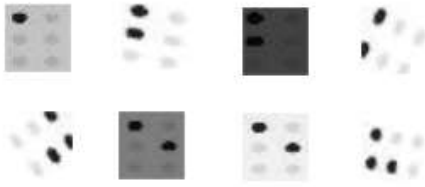


Figure 4. Exemplo da Diversidade do dataset

Na etapa 2, pré-processamento, as imagens de entrada foram redimensionadas para 320x320 *pixels* e foi criado dois novos *dataset*, a partir da análise das imagens do original, aplicando um ajuste na rotação das imagens de até 45 graus, para um dos *datasets*.

Na etapa 3, a arquitetura do modelo selecionado foi YOLOv5 (*You Only Look Once* versão 5), variação Small (YOLOv5s) que apresenta 3 subdivisões principais da rede: o (i) *Backbone*, responsável por extrair características (*features*) da imagem por convolução múltipla e agrupamento, (ii) a *Neck*, a qual trata essas *features* para a etapa de detecção, agregando e combinando as mesmas e (iii) o *Head*, que é responsável pela parte final da detecção, possuindo como saída um vetor contendo os dados das detecções feitas (descrições das caixas delimitadoras) e, com isso, a imagem com a localização dos objetos de interesse é formada, como mostra a Figura 5.

Na etapa 4, ocorre o ajuste dos principais hiperparâmetros utilizados na arquitetura convolucional do YOLOv5, sendo definidas a taxa de aprendizado de 0.01, momento igual a 0.937 e decaimento de peso (*weight decay*) de 0.0005. Em seguida, definiu-se o *batchsize* (tamanho de lote) igual a 16, com número de épocas igual a 200. Para a otimização da função de perda, foi escolhido o otimizador SGD (*Stochastic Gradient Descent*). Vale destacar que os ajustes dos hiperparâmetros foram realizadas por meios de testes de *accuracy* levando em consideração valores já definidos na literatura.

Por fim, na etapa 5, é apresentada as detecções dos caracteres Braille. As caixas delimitadoras (*bounding boxes*) apresentam uma pontuação na sua parte superior com uma medida de confiança do modelo, de que o objeto detectado, é de uma determinada classe. As caixas delimitadoras com pontuações de detecção abaixo de um determinado limiar são descartadas.

Configurações

A etapa de treinamento foi realizada na plataforma online Google Colab³, a qual disponibiliza uma máquina virtual com placa de vídeo dedicada (GPU), NVIDIA Tesla T4 12GB. A máquina local utilizada é composta por um processador Intel Core i7 de 7ª geração, 64 bits e 16GB de memória RAM rodando no sistema operacional Ubuntu 22.04 LTS.

Para o processamento dos dados, foi necessário definir as *labels*, previamente, num arquivo de extensão ".txt" com a classe do algoritmo - entende-se como classe a letra que o caractere pertence, como "a" ou "b", por exemplo - e as quatro coordenadas que formam a caixa delimitadora para o modelo

³<https://colab.research.google.com>

compreender em que parte da imagem está o caractere que deve ser aprendida. Como no *dataset* original tem apenas as imagens, as *labels* foram adicionadas com a ajuda da aplicação "LabelImg" [11].

O tutorial oficial disponível do YOLOv5 no Google Colab [12] foi utilizado nas etapas de treinamento e validação, executando o bloco de código disponível no "SETUP". O arquivo do modelo YOLOv5s pré-treinado foi utilizado como ponto de partida dos pesos iniciais para o modelo a ser treinado.

Métricas de Avaliação

Para avaliar este trabalho, utilizou-se a matriz de confusão e as métricas de desempenho precisão (*precision*), *recall* (sensibilidade), *F1-score* e mAP@.5. Precisão é a proporção de Verdadeiro Positivo classificados corretamente; *Recall* é a porcentagem de instâncias positivas que são classificadas como positivas, em outras palavras, o quão bom o classificador é para classificar corretamente a classe de interesse; *F1-score* é a média harmônica ponderada entre a precisão e *recall*, conforme apresentado nas Equações 1, 2, 3, respectivamente:

$$Precisão = \frac{VP}{(VP + FP)} \quad (1)$$

$$Recall = \frac{VP}{(VP + FN)} \quad (2)$$

$$F1 - Score = 2 \cdot \frac{Precisão \cdot Recall}{(Precisão + Recall)} \quad (3)$$

sendo VP, FP e FN os números de Verdadeiros Positivos, Falsos Positivos e Falsos Negativos, respectivamente.

Já o mAP (*mean average precision*) é uma métrica usada em detecção de objetos que representa a média da AP (*average precision*). Para este cálculo, é preciso calcular a Interseção sobre a União (IoU - do inglês *Intersection over Union*), cujo o processo pode ser observado na Figura 6. Neste trabalho, utilizou-se mAP@.5 (*mean Average Precision at IoU 0.5*), que significa que apenas os objetos corretamente previstos em caixas delimitadoras com IoU superior a 0,5 são considerados positivos. O mAP@.5 é utilizado como uma medida de quão bem o modelo é capaz de localizar e classificar os objetos corretamente em uma imagem.

RESULTADOS E DISCUSSÕES

As subseções a seguir descrevem os *datasets* e os resultados experimentais do modelo. Os experimentos foram realizados utilizando o conjunto de métricas de desempenho, apresentados na seção anterior, para avaliar a eficácia da abordagem proposta. A matriz de confusão foi utilizada por ser uma técnica que resume o desempenho do algoritmo de classificação, fornecendo uma análise mais detalhada sobre quais classes o modelo está conseguindo acertar ou errar [2].

Descrição do Dataset

Para verificar o impacto no desempenho proveniente do *dataset*, foram realizados vários experimentos usando o *dataset*

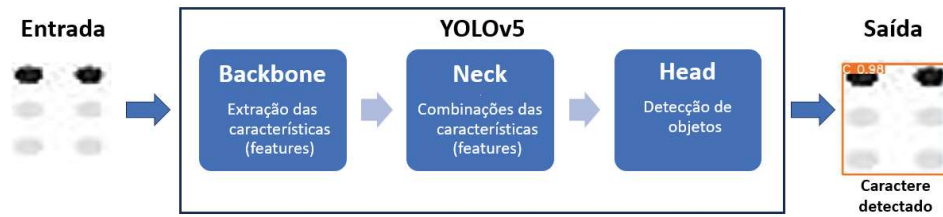


Figure 5. Arquitetura do modelo escolhido. Adaptado de [5, 1]

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} = \frac{\text{Área de Sobreposição}}{\text{Área da União}}$$

Figure 6. Representação do IoU [8]

original, apresentados na Tabela 1. No *dataset* original, *kaggleDataset*, foi gerado mais um *dataset*, *smoothRotation*, formados com caracteres com rotações suaves (de até 45 graus). Os *datasets* foram divididos em 70% para treino, 20% para validação e 10% para teste.

Métodos	Treino	Validação	Teste	Total
kaggleDataset	1.092	312	156	1.560
smoothRotation	806	260	130	1.196

No *kaggleDataset*, todo o *dataset* original foi mantido e apenas separado em treino, teste e validação. Contudo, analisando o *dataset*, foi verificado que existem diversas imagem com rotações abruptas (superiores a 45 graus), como as demonstradas na Figura 8(a), com isso, gerando ruídos no treinamento do modelo, ou seja, um caractere pode ser confundido com outro, devido esse acentuado grau de rotação.

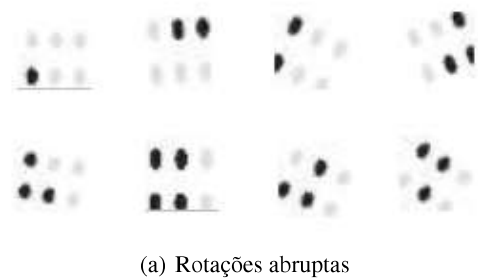
Ao analisar o *kaggleDataset*, observou-se que as rotações de um modo geral desempenham um papel influente na detecção e tradução de caracteres Braille. Rotações excessivas podem afetar significativamente a interpretação dos caracteres. Um exemplo disso pode ser percebido na Figura 7. Quando uma cela Braille representando o caractere “B” possui uma rotação abrupta de 90 graus, sua configuração se assemelha ao caractere Braille “C”, ocasionando o ruído no treinamento. Diante disso, retirar esse tipo de rotação tende a melhorar o desempenho do modelo e, para isso, foi gerado mais dois *datasets*, *smoothRotation* e *noRotation*, a partir do *kaggleDataset* para realizar esta análise.

Pelo fato do *dataset* original, disponível no Kaggle, não possuir um padrão de angulação regular, as imagens foram separadas manualmente. Assim, no *dataset* original, cada caractere possui 20 imagens com rotações variadas, de um total de 60 imagens por caractere; no *dataset smoothRotation* são 6 imagens com rotações leves para cada caractere, de um total de

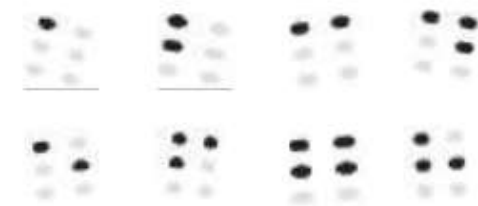


Figure 7. Exemplo de ruído gerado por rotação abrupta de caractere Braille

46 imagens por caractere, algumas que foram selecionadas podem ser verificadas na Figura 8(b).



(a) Rotações abruptas



(b) Rotações suaves

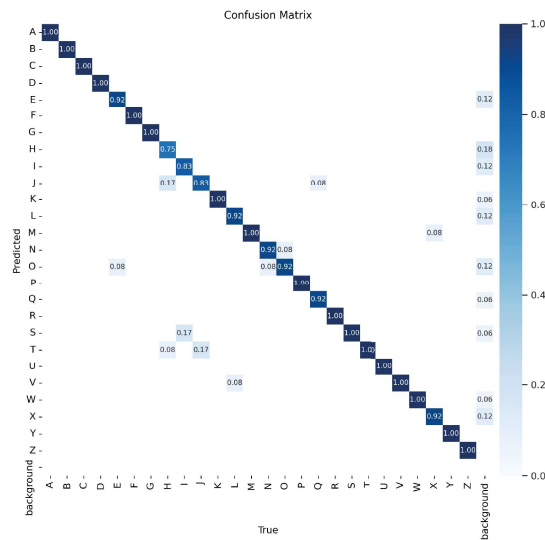
Figure 8. Exemplos de rotações no dataset original

Resultados

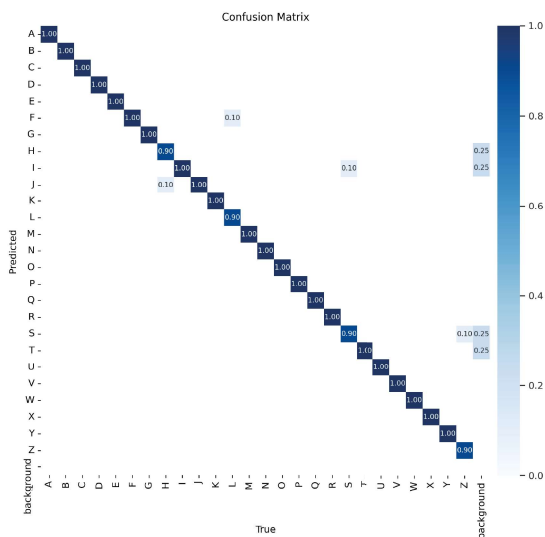
Após o treinamento completo utilizando diferentes configurações de parâmetros aplicados sobre os *datasets*, é apresentado na Figura 9, a Matriz de Confusão para cada *dataset* usado nos experimentos. As linhas representam as classes previstas pelo modelo (*Predicted*) e as colunas representam as classes verdadeiras (*True*), de modo que a predição correta para cada classe se encontra na diagonal principal da matriz, cuja a barra lateral à direita da matriz, indica a variação de valores que oscila de 0 a 100.

Através da matriz de confusão (Figura 9), é possível analisar de maneira mais detalhada os resultados obtidos pelo modelo.

A partir da análise experimental, é possível constatar que rotações bruscas (acima de 45 graus) indicam quedas consistentes na precisão de classificação, como é possível observar na Figura 9(a). O impacto de rotações mais condizentes com a realidade, inferiores a 45 graus, pode ser observado na Figura 9(b). Assim, pode-se notar que o *dataset* com rotações suaves melhora significativamente a precisão de reconhecimento, quando comparado ao *dataset* original.



(a) Método KaggleDataset



(b) Método smoothRotation

Figure 9. Matriz de Confusão.

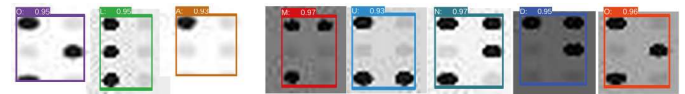
Na Tabela 2, pode-se verificar as métricas provenientes das matrizes de confusão, como Precisão, *Recall* e *F1-score* e *mAP@.5*. Esses resultados reforçam a capacidade do modelo YOLOv5 em reconhecer células Braille e validam sua aplicabilidade na conversão de caracteres.

O modelo YOLOv5, em sua arquitetura, apresenta a função *Focal Loss* - função que atribui maior peso aos exemplos difíceis, melhorando a capacidade de detecção e o desempenho

Table 2. Resultados dos Métodos

Métodos	Precision	Recall	F1 Score	mAP50
kaggleDataset	0.973	0.969	0.970	0.992
smoothRotation	0.990	0.985	0.990	0.995

geral do modelo - e a estratégia de treinamento em várias escalas. Dessa forma, na Figura 10, é apresentado exemplos de detecções e tradução realizada, sendo que para cada detecção, tem-se a caixa delimitadora, o caractere em português e a taxa de confiança da detecção.



(a) Ola mundo

Figure 10. Reconhecimento e tradução do caractere Braille

De modo geral, o modelo apresentou alto desempenho de reconhecimento Braille, com destaque para o *dataset smoothRotation*, por apresentar rotações mais condizentes com o que é praticado, por retratar as imagens como são capturadas, normalmente, por um dispositivo móvel. Além de apresentar resultados próximos do *noRotation*, o que demonstra a eficácia do modelo.

CONSIDERAÇÕES FINAIS

O sistema Braille desempenha um papel crucial para alunos com DV e para aqueles que estão próximos deles. Ao introduzir recursos tecnológicos, como reconhecimento Braille, estes têm o potencial de causar um impacto significativo na comunidade que depende do Braille. Essas inovações permitem que o Braille seja amplamente interpretado e compreendido, abrindo portas para uma comunicação mais inclusiva e facilitando a participação plena desses indivíduos na escola e sociedade.

Dessa forma, a principal contribuição deste trabalho foi desenvolver um método para detectar os caracteres Braille e apresentar a sua correspondente em português com precisão, de modo que possa vir facilitar a comunicação escrita entre os professores videntes e os alunos com DV.

Agradecimento

O primeiro autor é bolsista do CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil.

REFERENCES

- [1] Davi Albuquerque, Antonio Braga, Isac Bomfim, and Danielo Gomes. 2022. Aplicando um modelo YOLO para detectar e diferenciar por imagem castas de abelhas melíferas de forma automatizada. In *Anais do XIII Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais*. SBC, Porto Alegre, RS, Brasil, 51–60. DOI: <http://dx.doi.org/10.5753/wcama.2022.222918>
- [2] Jason Brownlee. 2016. What is a Confusion Matrix in Machine Learning. Disponível em: <https://machinelearningmastery.com/>

- confusion-matrix-machine-learning/**. (2016). Acessado em 02 de julho de 2023.
- [3] Bi Min Hsu. 2020. Braille recognition for reducing asymmetric communication between the blind and non-blind. *Symmetry* 12, 7 (2020). DOI : <http://dx.doi.org/10.3390/SYM12071069>
- [4] IlyaOvodov. 2020. AngelinaDataset. Disponível em: <https://github.com/IlyaOvodov/AngelinaDataset/blob/master/books/mdd-redmi103.labeled.jpg>. (2020). Acessado em 02 de junho de 2023.
- [5] Elżbieta Kubera, Agnieszka Kubik-Komar, Paweł Kurasinski, Krystyna Piotrowska-Weryszko, and Magdalena Skrzypiec. 2022. Detection and Recognition of Pollen Grains in Multilabel Microscopic Images. *Sensors* 22, 7 (mar 2022), 2690. DOI : <http://dx.doi.org/10.3390/s22072690>
- [6] Helena Leite. 2021. Inclusão na pandemia: como crianças surdas e cegas se adaptaram às aulas online. Disponível em: <https://paisefilhos.uol.com.br/familia/inclusao-na-pandemia-criancas-surdas-e-cegas-contam-como-se-adaptaram-as-aulas-online/>. (2021). Acessado em 8 de julho de 2023.
- [7] Ilya G. Ovodov. 2021. Optical Braille Recognition Using Object Detection Neural Network. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 1741–1748. DOI : <http://dx.doi.org/10.1109/ICCVW54120.2021.00200>
- [8] Rafael Padilla, Sergio L. Netto, and Eduardo A. B. da Silva. 2020. A Survey on Performance Metrics for Object-Detection Algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 237–242. DOI : <http://dx.doi.org/10.1109/IWSSIP48289.2020.9145130>
- [9] Shanks0465. 2020. Braille Character Dataset. Disponível em: <https://www.kaggle.com/datasets/shanks0465/braille-character-dataset>. (2020). Acessado em 2 de junho de 2023.
- [10] Jevri Tri Ardiansah and Yasuhisa Okazaki. 2020. The Design and Prototyping of Braille to Speech Application as a Self-Learning Support Media for Visually Impaired Person. In *2020 4th International Conference on Vocational Education and Training (ICOVET)*. IEEE, 224–228. DOI : <http://dx.doi.org/10.1109/ICOVET50258.2020.9230060>
- [11] Tzatalin. 2015. LabelImg. Disponível em: <https://github.com/heartexlabs/labelImg>. (2015). Acessado em 2 de junho de 2023.
- [12] Ultralytics. 2022. YOLOv5 Tutorial. Disponível em: <https://colab.research.google.com/github/ultralytics/yolov5/blob/master/tutorial.ipynb>. (2022). Acessado em 2 de junho de 2023.
- [13] Shanmuganathan Vasanthapriyan and Malith De Silva. 2019. Optical Braille Recognition Software Prototype for the Sinhala Language. *Advances in Science, Technology and Engineering Systems Journal* 4, 4 (2019), 221–229. DOI : <http://dx.doi.org/10.25046/aj040427>
- [14] Ivani Cristina Voos and Fábio Peres Gonçalves. 2019. O Desenvolvimento Profissional de Docentes da Educação Especial e o Ensino de Ciências da Natureza para Estudantes Cegos e Baixa Visão. *Revista Brasileira de Educação Especial* 25, 4 (dec 2019), 635–654. DOI : <http://dx.doi.org/10.1590/s1413-65382519000400007>