SEGMENTING FRESHWATER FISH IMAGES WITH CONVOLUTIONAL NEURAL NETWORKS

Nicolas Figueiredo Cavalcante Sales
Departamento de Ciência da Computação
Universidade Federal de Rondônia
Porto Velho, Brasil
nicolas.cavalcante.dev@gmail.com

Carolina Yukari Veludo Watanabe
Departamento de Ciência da Computação
Universidade Federal de Rondônia
Porto Velho, Brasil
carolina@unir.br

***Abstract*—The acquisition of important information from fisheries for studies in Ichthyology and Fisheries currently takes place manually. Data on the length and morphology of freshwater fish are valuable for researchers to determine indicators in their studies. Technological development in this area aims to increase the speed and effectiveness of information gathering, assisting researchers, students, and fishermen in this goal. Aligned with this objective, this work presents a comparison between two models, Mask R-CNN and YOLOv8, used to segment fish images and generate their masks, ultimately feeding them into an automatic measurement algorithm, which is the broader purpose of this project. The results show that the models can segment various types of fish in different positions and environments, with Mask R-CNN achieving 80.16% using the IOU (Intersection over Union) metric and YOLOv8 achieving 86.15%.**

Keywords—Image segmentation, Deep Learning, Convolutional Neural Networks, Freshwater fish image segmentation, Mask R-CNN, YOLO, YOLOv8.

I. INTRODUÇÃO

Na área da Ictiologia, existem vários indicadores importantes para análise da fauna pesqueira na bacia amazônica. De acordo com Rocha [8], essas informações podem ser usadas para levantar vários tipos de pesquisas, desde relacionados ao impacto das ações humanas nas comunidades pesqueiras até a observação da maturidade e do tamanho do peixe obtido para algum tipo de indicador populacional. No entanto, a obtenção destas informações ainda são feitas comumente por formas manuais, requisitando um conhecimento técnico de uma pessoa para sua realização. Por exemplo, para a medição do comprimento e largura do peixe, é usado uma fita métrica por um operador humano que, ao fazer a ação, guarda as informações em um dispositivo ou em um caderno. Isso, em uma situação real, pode acarretar erros além do fato de necessitar de um longo período de tempo para sua execução, dificultando o processo e limitando a qualidade e quantidade dos dados obtidos. Desta forma, levantar um sistema que consiga realizar a obtenção de informações de maneira automática oferece maior velocidade em todo o processo, reduz a possibilidade de erros em adição de se tornar mais escalável e vantajoso para indivíduos sem treinamento especializado [8].

Nesse contexto, as técnicas de processamento de imagens e os modelos de inteligência artificial estão se tornando soluções concretas para satisfazer com o objetivo, pois permitem uma abordagem rápida para a coleta de dados da pesca [8]. E, para isso, a extração de contornos de objetos a partir de imagens tornaram-se a base de muitas aplicações com esse fim, já que, com o contorno do peixe, podemos utilizar técnicas para retirar informações, tal como o seu comprimento.

Essa obtenção de contornos de objetos em imagens é um método denominado de segmentação que, neste trabalho, foi uma responsabilidade executada pelas redes neurais convolucionais (do inglês, *Convolutional Neural Networks* – CNN), que são algoritmos usados para tarefas relacionados a análise de imagens e visão computacional [1].

No trabalho feito por [7], foi desenvolvido um projeto com o uso do modelo Mask R-CNN [3] para realizar a segmentação dos peixes e em seguida sua medição morfológica, obtendo bons resultados em geral com um indicador IOU de 82.21%, focando nos índices apenas relacionado ao modelo. Apesar de os resultados apresentados terem sido satisfatórios em geral, a pesquisa nessa área não finalizou, ficando em aberto o uso de diferentes técnicas para segmentação e medição de peixes, buscando melhores resultados e contribuindo para o aprimoramento do projeto.

Assim sendo, o objetivo deste trabalho foi estudar e implementar redes neurais convolucionais para a tarefa de segmentação de imagens de peixes. Como objetivos específicos, teve-se:

1. Estudar e implementar redes neurais convolucionais apresentados na literatura adequados para segmentação de peixes (de águas marítimas);

2. Adequar os métodos implementados para atenderem às especificidades dos tipos de pescados da Amazônia;

3. Integrar o melhor método desenvolvido ao aplicativo ICTIOBIOMETRIA.

Para isso, foram estudados dois modelos de inteligência artificial para segmentação de peixes de água doce em imagens, o Mask R-CNN [3] usando a biblioteca Detectron2 do Facebook AI Research [11] e o YOLOv8 da Ultralytics [4], visando usar implementações atuais e comparando-os com os resultados do modelo treinado por Rocha [8] [7], a fim de propor novos algoritmos mais eficientes e precisos para a tarefa de segmentação. As imagens usadas foram adquiridas junto ao Laboratório de Ictiologia e Pesca, de peixes da bacia do Rio Madeira, Porto Velho, Rondônia, que faz parte da Amazônia Ocidental, com ajuda de biólogos especialistas na área.

Este trabalho está estruturado da seguinte maneira. Na seção 2, será explicado e apresentado os modelos de redes neurais convolucionais Mask R-CNN e YOLOv8, tal como as ferramentas, técnicas, conceitos e métodos utilizados neste trabalho. A seção 3 apresenta os resultados obtidos

relacionando-os com o trabalho de Rocha [8]. E, por último, a seção 4 apresenta as conclusões do projeto.

II. METODOLOGIA

Nesta seção serão explicados os conceitos de uma CNN e dos modelos Mask R-CNN e YOLOv8, além disso, será apresentado as etapas para execução dos métodos propostos.

A. CNN

Uma CNN é um modelo específico de rede neural utilizado para tarefas de visão computacional [8] [1]. Rede neural é um algoritmo composto de um conjunto de operações matemáticas que podem ser treinadas com dados buscando um padrão estatístico para responder e prever resultados de acordo com uma entrada [1]. Nas CNNs, a camada de convolução desempenha um trabalho fundamental no funcionamento da rede, executando um conjunto de algoritmos e cálculos algébricos nas matrizes das imagens a fim de obter valores importantes para a rede e para o aprendizado. Uma CNN pode conter um número variado de camadas de convolução, dependendo da complexidade da imagem e do modelo proposto [8]. Quando uma imagem é inserida na rede ela passa pelas camadas de convolução, o resultado obtido a partir desse processo é denominado como mapa de característica, onde comporta as principais características da imagem, possibilitando que a rede neural consiga reconhecer os objetos dentro da imagem.

B. Mask R-CNN

Mask R-CNN é uma rede de propósito geral para tarefas relacionadas à segmentação de objetos. Seu propósito é levantar um método de detecção de imagem ao mesmo tempo que ocorre a sua segmentação, resultando em uma máscara em alta qualidade de cada instância. Esse modelo é uma extensão ao Faster R-CNN adicionando funcionalidades para predição de máscaras de objetos em paralelo com a existente parte de criação de uma *bounding box* de reconhecimento para as respectivas instâncias [3].

O seu funcionamento dessa rede neural ocorre em etapas como visualizado na Figura 1.

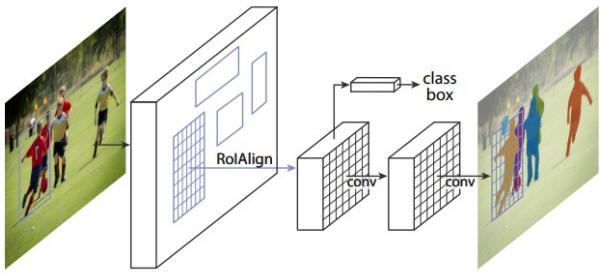

Fig. 1. Algoritmo do Mask R-CNN. Fonte: [3].

No início de seu processo, a entrada é uma imagem que é inserida em uma rede ResNext101 ─ que é uma rede neural convolucional com 101 camadas ─ e FPN (*Feature Pyramid Network*) construída e treinada para obtenção do mapa de característica. Em seguida, a rede vai inserir um número de ROIs (*Region of Interest*) determinadas pelo pesquisador na imagem. ROI é uma caixa delimitadora que identifica um objeto candidato, que pode ser ou não identificado como um objeto de interesse para a rede. Ao determinar as ROIs, o RPN (*Region Proposal Network*) determinará se em cada ROI contém um objeto desejado, caso sim, retorna as coordenadas da instância. Após isso, é executada a operação de *ROIAlign* nas ROIs escolhidas a fim de alinhar as características obtidas, levantando uma maior acurácia nas caixas delimitadoras. Por último, com as extrações das ROIs, são executadas as operações convolucionais da rede até o processo de classificação e identificação dos objetos com suas devidas máscaras retornadas [8].

C. YOLOv8

A primeira versão do modelo denominado YOLO foi introduzido em um repositório da linguagem C denominado Darknet no ano de 2015 por Joseph Redmon [6], desde então, a comunidade vem desenvolvendo novas versões. Desenvolvido pela equipe da Ultralytics, o YOLOv8 é usado na detecção, classificação e segmentação de objetos em imagens [4]. A equipe vem realizando um conjunto de melhorias do modelo, fazendo-o ficar melhor e mais fácil de usar que um de seus produtos anteriores, o YOLOv5. O YOLOv8 é um modelo avançado que levanta aprimoramentos e modificações incluindo uma diferente rede *backbone* e uma nova função de perda [9].

Visando agora entender o processo de execução dos métodos apontados como exemplificado na Figura 2, a primeira etapa que é necessário realizar é o trabalho de obtenção de imagens de peixe. Como estamos trabalhando com redes neurais convolucionais para tarefas de visão computacional, imagens são a nossa principal fonte. Desta forma, quanto mais imagens de peixe de água doce conseguimos, melhores resultados obtemos. A base de imagens foi colhida graças aos pesquisadores e biólogos do Laboratório de Ictiologia e Pesca da Universidade Federal de Rondônia (UNIR) com a participação de pesquisadores do Departamento de Ciência da Computação também da mesma universidade. As imagens consistem de fotos retiradas do mercado de peixes da cidade de Porto Velho, Rondônia, como observado na Figura 3, além de um pequeno número ter sido coletado por fotos de dentro do laboratório de Ictiologia.

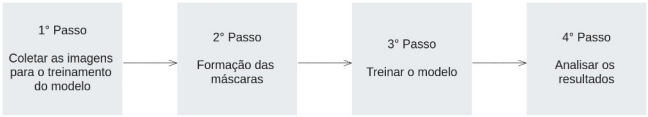

Fig. 2. Fluxograma da execução geral do método proposto.

Fig. 3. Fotos de peixes.

As fotos foram retiradas diretamente por *smartphones* dos pesquisadores e conseguimos observar que só nos exemplos levantados, podemos encontrar diferentes espécies de peixes. Neste trabalho, foi usado um total de 219 imagens que para um treinamento de uma rede neural, é pouco. Tendo em vista esse fato, foi necessário realiazar um passo extra, tendo que executar uma técnica chamada *data augmentation*, que objetiva a expansão de uma base de imagens pelo resultado de um conjunto de operações que modificam as amostras originais, como rotações, espelhamentos, ajustes de brilho e contraste, entre outras, isso para gerar novas variações das imagens, aumentando assim a diversidade da nossa base e melhorando a capacidade de generalização do modelo treinado. Depois da execução do método, conseguimos

aumentar de 219 para 588 imagens, que já é um número interessante para irmos à próxima etapa.

Com as imagens, temos agora que criar as máscaras dos peixes. Esse processo visa mostrar o modelo a forma em como queremos segmentar os objetos dentro de uma imagem, já que estamos tratando com uma rede supervisionada que necessita de informações etiquetadas de posição e forma dos objetos que queremos que ela aprenda a segmentar ─ podemos observar um exemplo de máscara na Figura 4. Para esse processo, utilizamos o *software* Labelme [10], onde manualmente devemos criar as máscaras de cada objeto dentro da imagem como observado na Figura 5.

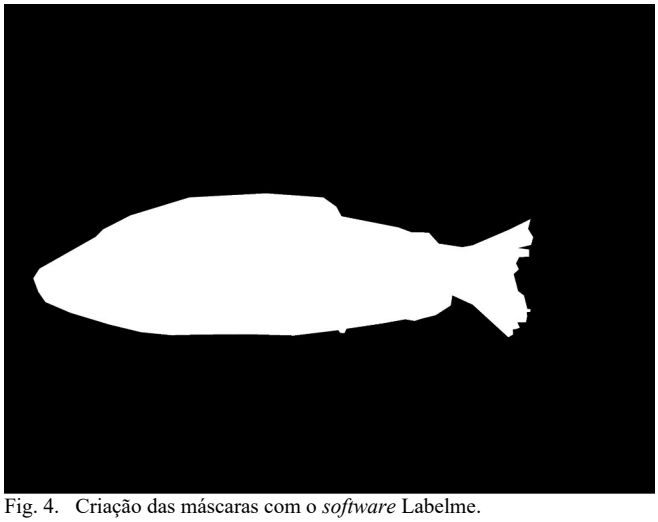

Fig. 4. Criação das máscaras com o *software* Labelme.

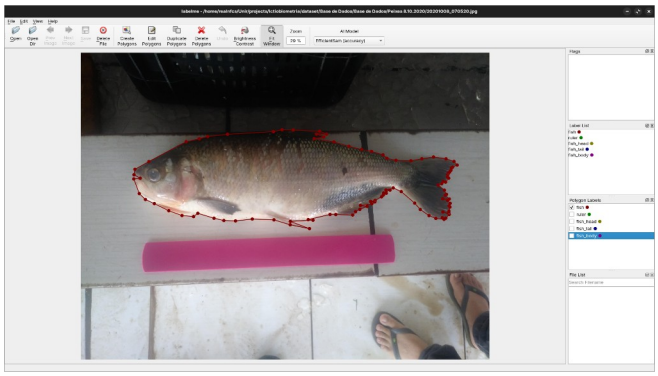

Fig. 5. Criação das máscaras com o *software* Labelme.

Com as imagens e suas respectivas máscaras criadas, podemos então iniciar o treinamento dos modelos. Tanto o YOLOv8 do Ultralytics como o Mask R-CNN do Detectron2 possuem suas próprias documentações para o treinamento com dados customizados. Neste trabalho, utilizamos somente uma classe de identificação dos objetos, denominado *fish* (do inglês, peixe), agrupando todas as categorias de pescado nessa única classe. Utilizamos um total de 567 imagens para toda execução do treinamento onde, após esse processo, os modelos ficam disponíveis para o processo de inferência que é ação de aplicar em um modelo treinado novos dados para realizar previsões ou gerar saídas que, no nosso trabalho, é retornar a posição da máscara do peixe dentro da imagem, podendo agora analisar os resultados e validá-los.

III. Resultados E Discussões

Para execução do treinamento do modelo YOLOv8 da Ultralytics [4], foi utilizado um ambiente com uma placa de vídeo NVIDIA Quadro P620, com o processador Intel Core i7-8700 e 16GB de RAM. Como parâmetros, foi usado um total de 100 épocas com o tamanho de imagem de entrada de 640 para respeitar com os requerimentos da rede e foi usado um modelo pré-treinado para segmentação com pesos relacionados ao *dataset* do COCO [5]. As outras configurações de parâmetros permaneceram na forma padrão definido pela biblioteca, ademais, para sua efetiva execução, usamos os comandos de terminal disponibilizado pela própria biblioteca do YOLO.

Já para o Mask R-CNN do Detectron2 [11], foi utilizado a plataforma Google Colab pelo navegador, que permite executar algoritmos na linguagem Python, a linguagem de programação escolhida para a implementação dos métodos. Na plataforma, é disponibilizado diferentes tipos de GPU e *runtimes* de forma paga e gratuita; no caso, foi utilizado o *runtime* T4 GPU que possui uma GPU NVIDIA Tesla T4. Depois da devida configuração do banco de imagens, os parâmetros definidos para rede foram o valor 2 para o número de processos paralelos a serem utilizados para carregar os dados, também o valor 2 para a quantidade de imagens que são processadas por iteração, 0.00025 como valor para a taxa de aprendizado, 1000 iterações que a rede executará, 256 sendo o número de amostras para propostas de região ou ROI por imagem e definindo o valor de classes como 1 pois só temos um objeto a ser segmentado; as outras configurações permaneceram o padrão definido pela biblioteca. Além disso, utilizamos o modelo do Mask R-CNN disponibilizado com pesos pré-treinados pelo *dataset* do COCO [5] com a estrutura de *backbone* FPN, para a devida predição de máscaras e de caixas delimitadoras.

A métrica usada para avaliação dos resultados levantados pelos modelos chama-se IOU (1). Como nos resultados obtivemos o IOU de um conjunto de máscaras de saída, como métrica geral usamos a média (2) para determinar a eficiência e precisão dos modelos.

$$IoU = \frac{Área\ \ da\ \ Intersecção}{Área\ \ da\ \ União} \tag{1}$$

$$Média\ \ do\ \ IoU = \frac{\sum_{i=1}^{N} IoU_i}{N} \tag{2}$$

Para o efetivo cálculo de (1), necessitamos da máscara original e a máscara de saída que o modelo infere nas formas binarizadas. Com isso, realizamos a intersecção e a união das máscaras e em seguida calculamos a sua razão e multiplicamos por 100, obtendo um índice que determina o quão próximo a saída chegou do esperado. Depois dos treinamentos, realizamos a inferência em 21 imagens de teste que não foram usadas no processo, isso significa que são informações novas para os modelos. O melhor índice IOU do YOLOv8 em uma imagem foi de 97.46% como observado na Figura 6, enquanto o pior foi de 22.51%, levantado na Figura 7. Já no caso do Mask R-CNN, o melhor índice foi de 95.62% (Figura 8) e o pior com 23.05% (Figura 9).

Fig. 6. Melhor métrica IOU no processo de inferência do YOLOv8.

Fig. 7. Pior métrica IOU no processo de inferência do YOLOv8.

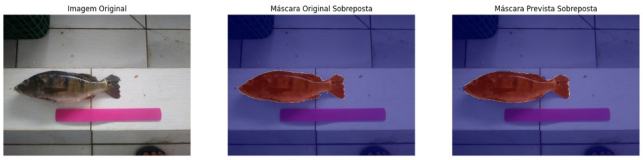

Fig. 8. Melhor métrica IOU no processo de inferência do Mask R-CNN.

Fig. 9. Pior métrica IOU no processo de inferência do Mask R-CNN.

Pelas imagens (Figuras 6-9), conseguimos visualizar os acertos e erros dos modelos. O YOLOv8 conseguiu de maneira geral segmentar peixes de maneira satisfatória, porém não conseguiu segmentar todos os peixes em uma imagem, somente alguns. No caso do Mask R-CNN, em relação às inferências com resultados ruins, percebemos que ocorreu a segmentação a mais do que o esperado do peixe, além de ter sido retornada a segmentação de outros objetos que não queríamos no resultado final. Esses fatos mostram que ambos os modelos conseguem de maneira satisfatória segmentar as amostras quando esses estão isolados nas imagens, porém, no caso do YOLOv8, ele não consegue segmentar mais de um peixe na imagem, e em relação ao Mask R-CNN, se tiver um pedaço de um outro peixe muito perto ou junto ao peixe alvo da inferência, ele tende a segmentar ambos entendendo que os dois são um só.

Essas foram métricas IOU individuais retiradas das inferências das 21 imagens de teste, mas se quisermos um índice geral para determinar a precisão dos modelos, usamos a média levantada em (2), ques corresponde à a soma de todas as métricas IOU de cada máscara retornada dos modelos, dividindo pelo total de imagens de teste e em seguida multiplicando o resultado final por 100 para obtemos a porcentagem de acerto geral do modelo. Como observado na Tabela 1, o YOLOv8 conseguiu uma métrica de 86.15%, enquanto o Mask R-CNN alcançou 80.16%.

TABLE I. RESULTADOS ATIGINDOS DOS TREINAMENTOS

Modelo	***Métrica IOU***
YOLOv8	86.15%
Mask R-CNN	80.16%

Conseguimos observar que neste trabalho, o YOLOv8 conseguiu um melhor resultado que o outro modelo.

Uma situação adicional a citar é que, comparando os resultados com o trabalho realizado por [7] que utilizaram o modelo Mask R-CNN [3], percebemos a melhora que teve os modelos atuais na tarefa de segmentação de peixes, já que o YOLOv8 para a mesma tarefa conseguiu uma métrica IOU maior que os 82.21% conquistados pelos autores com a rede, e mesmo o Mask R-CNN do Detectron2 [11] que conseguiu uma métrica abaixo da citada, ele ainda torna-se mais interessante que o modelo usado por [7], principalmente por conta de sua velocidade e facilidade de configuração e execução que, se treiná-lo com mais de 1000 iterações, poderá passar a métrica dos autores.

Outro fato que contribui para os melhores resultados dos modelos deste trabalho em relação ao usado por [7] é a diferença no número de amostras usados para o treinamento das redes, já que os autores utilizaram um total de 378 imagens, enquanto neste trabalho foram utilizadas 567 imagens, situação que confirma que o aumento de amostras seguida com uma boa configuração de parâmetros do modelo levanta uma boa rede para a tarefa de segmentação.

IV. CONCLUSÕES

Neste trabalho foi explorado o uso de dois modelos de redes neurais convolucionais para a tarefa de segmentação de peixes de água doce, usando os modelos YOLOv8 da Ultralytics e o Mask R-CNN da biblioteca Detectron2 da Facebook AI Research. A partir dos resultados obtidos depois do efetivo treinamento dos modelos, pode-se observar que estas duas redes apresentaram resultados bastante promissores para a segmentação, já que o primeiro obteve uma métrica IOU com média geral de precisão de 86.15%, enquanto o segundo alcançou o índice de 80.16%.

Para trabalhos futuros, é essencial expandir a base de dados, incluindo imagens capturadas tanto em ambientes controlados (como laboratórios) quanto em cenários reais (como mercados de peixe). Essa ampliação ajudará a capturar variações nas condições de iluminação, posicionamento e contextos visuais, aspectos que influenciam diretamente o desempenho do modelo na segmentação. Observou-se, por exemplo, que o modelo apresenta dificuldade ao segmentar peixes em proximidade de outros peixes ou objetos, além de desafios com espécies menos representadas na base de dados. Para abordar esses desafios, será importante também investigar arquiteturas alternativas e ajustar parâmetros diversos, o que pode revelar configurações que maximizem a qualidade da segmentação. Posteriormente, realizar o retreinamento dos modelos, seja a partir do zero ou utilizando a técnica de *fine-tuning* [1], permitirá uma adaptação mais eficaz a essas variações, aprimorando a precisão das inferências. Tais esforços enriquecem a literatura sobre o uso de redes neurais para segmentação de peixes de água doce e contribuem para o avanço da área.

Por fim, a separação dos resultados de inferência por espécie de peixe oferece uma perspectiva valiosa para avaliar a performance da inteligência artificial e deve ser considerada em futuros trabalhos. Com essa abordagem, torna-se possível identificar lacunas na base de dados relacionadas a espécies específicas, permitindo direcionar esforços na coleta de imagens de determinadas tipos de peixes da bacia amazônica. Esse aprimoramento visa aumentar a precisão e a versatilidade do modelo, garantindo uma segmentação mais eficaz e abrangente entre as diversas espécies da região.

REFERÊNCIAS

[1] CHOLLET, François. Deep learning with Python. Second edition. Shelter Island: Manning Publications, 2021.

[2] GOOGLE. Google Colaboratory. 2024. Disponível em: https://colab.research.google.com.

[3] HE, Kaiming et al. Mask R-CNN. Versão arXiv:1703.06870. [S. l.]: arXiv, 24 jan. 2018. arXiv:1703.06870 [cs]. Disponível em: http://arxiv.org/abs/1703.06870. Acesso em: 14 ago. 2024.

[4] JOCHER, Glenn; CHAURASIA, Ayush; QIU, Jing. Ultralytics YOLOv8. Versão 8.0.0. 2023. Disponível em: https://github.com/ultralytics/ultralytics.

[5] LIN, Tsung-Yi et al. Microsoft COCO: Common Objects in Context. Versão arXiv:1405.0312. [S. l.]: arXiv, 20 fev. 2015. arXiv:1405.0312 [cs]. Disponível em: http://arxiv.org/abs/1405.0312. Acesso em: 14 ago. 2024.

[6] REDMON, Joseph et al. You Only Look Once: Unified, Real-Time Object Detection. Versão arXiv:1506.02640. [S. l.]: arXiv, 9 maio 2016. arXiv:1506.02640 [cs]. Disponível em: http://arxiv.org/abs/1506.02640. Acesso em: 14 ago. 2024.

[7] ROCHA, Wan Song et al. Automatic measurement of fish from images using convolutional neural networks. Multimedia Tools and Applications, [s. l.], 18 abr. 2024. Disponível em: https://link.springer.com/10.1007/s11042-024-19180-1. Acesso em: 14 ago. 2024.

[8] ROCHA, Wan Song; DORIA, Carolina Rodrigues Da Costa; WATANABE, Carolina Yukari Veludo. Fish Detection and Measurement based on Mask R-CNN. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES, 2020, Brasil. Anais Estendidos da Conference on Graphics, Patterns and Images (SIBRAPI Estendido 2020). Brasil: Sociedade Brasileira de Computação, 7 nov. 2020. p. 183–186. Disponível em: https://sol.sbc.org.br/index.php/sibgrapi_estendido/article/view/13007. Acesso em: 14 ago. 2024.

[9] SOHAN, Mupparaju; SAI RAM, Thotakura; RAMI REDDY, Ch. Venkata. A Review on YOLOv8 and Its Advancements. In: JACOB, I. Jeena; PIRAMUTHU, Selwyn; FALKOWSKI-GILSKI, Przemyslaw (org.). Data Intelligence and Cognitive Informatics. Singapore: Springer Nature Singapore, 2024. (Algorithms for Intelligent Systems). p. 529–545. E-book. Disponível em: https://link.springer.com/10.1007/978-981-99-7962-2_39. Acesso em: 14 ago. 2024.

[10] WADA, Kentaro. Labelme: Image Polygonal Annotation with Python. [s. d.]. Disponível em: https://github.com/wkentaro/labelme.

[11] WU, Yuxin et al. Detectron2. 2019. Disponível em: https://github.com/facebookresearch/detectron2.