



PEDRO REGINATTO CONSTANTINO

**Uso de *Deep Learning* para predição de idades
de peixes em imagens de otólitos**

Maringá - Paraná
2026

PEDRO REGINATTO CONSTANTINO

Uso de *Deep Learning* para predição de idades de peixes em imagens de otólitos

Dissertação apresentada ao Programa de Pós-graduação em Bioestatística do Centro de Ciências Exatas da Universidade Estadual de Maringá como requisito parcial para obtenção do título de mestre em Bioestatística.

Orientador: Prof. Dr. Diego Corrêa Alves, Universidade Estadual de Maringá.

Membro Externo: Prof. Dr. Haroldo Valentin Ribeiro, Universidade Estadual de Maringá.

Membro Externo: Prof. Dr. Luiz Carlos Gomes, Universidade Estadual de Maringá.

Universidade Estadual de Maringá - UEM

Departamento de Estatística - DES

Programa de Pós-Graduação em Bioestatística

Maringá - Paraná

2026

Dados Internacionais de Catalogação-na-Publicação (CIP)
(Biblioteca Central - UEM, Maringá - PR, Brasil)

C758u

Constantino, Pedro Reginatto

Uso de *Deep Learning* para predição de idades de peixes em imagens de otólitos /
Pedro Reginatto Constantino. -- Maringá, PR, 2026.
45 f. : il. color., figs., tabs.

Orientador: Prof. Dr. Diego Corrêa Alves.

Dissertação (mestrado) - Universidade Estadual de Maringá, Centro de Ciências
Exatas, Departamento de Estatística, Programa de Pós-Graduação em Bioestatística,
2026.

1. *Deep Learning*. 2. Automação em ecologia pesqueira. 3. Ciência de dados - Pesca.
4. Peixes - Idade. 5. Otólitos - Imagens. I. Alves, Diego Corrêa, orient. II. Universidade
Estadual de Maringá. Centro de Ciências Exatas. Departamento de Estatística. Programa
de Pós-Graduação em Bioestatística. III. Título.

CDD 23.ed. 519.5

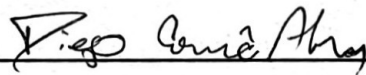
Elaine Cristina Soares Lira - CRB-9/1202

PEDRO REGINATTO CONSTANTINO

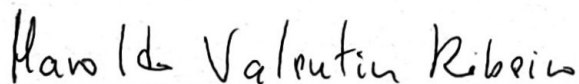
Uso de *Deep Learning* para predição de idades de peixes em imagens de Otólitos

Dissertação apresentada ao Programa de Pós-Graduação em Bioestatística do Centro de Ciências Exatas da Universidade Estadual de Maringá, como requisito parcial para a obtenção do título de Mestre em Bioestatística.

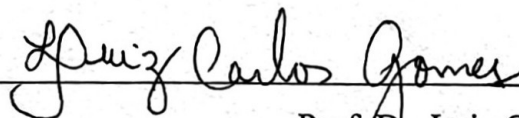
BANCA EXAMINADORA



Prof. Dr. Diego Corrêa Alves
Universidade Estadual de Maringá – PBE/UEM



Prof. Dr. Haroldo Valentin Ribeiro
Universidade Estadual de Maringá – PFI/UEM



Prof. Dr. Luiz Carlos Gomes
Universidade Estadual de Maringá – PEA/UEM



Maringá, 25 de fevereiro de 2026.

AGRADECIMENTOS

Expresso minha gratidão a todos que, de alguma forma, contribuíram para a realização deste trabalho.

Aos meus amigos do mestrado, aos professores, ao Departamento de Estatística e à Universidade Estadual de Maringá, pelo acolhimento e pelo apoio ao longo dessa caminhada.

Agradeço à CAPES pelo apoio financeiro. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

RESUMO

A estimativa da idade de peixes é uma importante etapa para a avaliação e manejo sustentável de estoques pesqueiros. Tradicionalmente, esse processo é realizado por especialistas por meio da análise visual de otólitos, o que demanda tempo, treinamento e está sujeito a variabilidade subjetiva. Nos últimos anos, estudos têm explorado o uso de técnicas de *deep learning*, mais especificamente redes neurais convolucionais, para automatizar a predição da idade de peixes a partir de imagens de otólitos. Neste trabalho, avaliou-se o desempenho de uma rede neural convolucional baseada na arquitetura InceptionV3 para a predição da idade de peixes a partir de imagens de otólitos. Foram investigadas estratégias de pré-processamento das imagens, bem como diferentes abordagens de treinamento visando mitigar o efeito do desbalanceamento da distribuição etária no conjunto de dados. O problema foi tratado como regressão, sendo o desempenho dos modelos avaliado por meio do erro quadrático médio, do coeficiente de determinação e da média do coeficiente de variação. Adicionalmente, analisou-se o impacto da aplicação de técnicas de *test-time augmentation* no desempenho preditivo. Os resultados indicam que determinadas estratégias de pré-processamento e balanceamento contribuem para melhorias no desempenho global do modelo. Contudo, observou-se pior desempenho preditivo para idades mais avançadas, possivelmente associado à baixa representatividade dessas faixas etárias no conjunto de dados.

Palavras-chave: aprendizagem profunda; reconhecimento de padrões; ciência de dados aplicada à pesca; automação em ecologia pesqueira.

ABSTRACT

Fish age estimation is an important step for the assessment and sustainable management of fish stocks. Usually, this process is performed by specialists through the visual analysis of otolith images, which demands time, training and is susceptible to reader bias. Over the past years, deep learning techniques have been explored, specifically convolutional neural networks, to automate fish age estimation by using otolith images. In this study, the performance of a convolutional neural network based on the InceptionV3 architecture was evaluated for predicting fish age from otolith images. Image preprocessing strategies were investigated, as well as alternative training approaches to mitigate the effects of age distribution imbalance in the dataset. The problem was formulated as a regression task, and model performance was assessed using the mean squared error, the coefficient of determination, and the mean coefficient of variation. Additionally, the impact of applying test-time augmentation on predictive performance was analyzed. The results indicate that some of the preprocessing and balancing strategies contribute to improvements in overall model performance. However, for older individuals, the models had a poor predictive performance, likely due to the low representation of those age classes in the dataset.

Key-words: deep learning; pattern recognition; data science applied to fisheries; automation in fisheries ecology.

LISTA DE ILUSTRAÇÕES

Figura 1 – Modelo de um neurônio k	14
Figura 2 – Exemplo de RNA.	15
Figura 3 – Gráficos de funções de ativação.	16
Figura 4 – Modelo de RNA com n neurônios de entrada e k neurônios de saída.	18
Figura 5 – Modelo de RNA com n neurônios de entrada, uma camada oculta com 5 neurônios e k neurônios de saída.	19
Figura 6 – Exemplo de convolução 2-D.	20
Figura 7 – Exemplos de módulo <i>Inception</i>	21
Figura 8 – Arquitetura InceptionV3.	22
Figura 9 – Exemplo <i>rembg</i>	23
Figura 10 – Exemplo de detecção de objetos utilizando YOLO.	24
Figura 11 – Coleta do otólito do halibute.	25
Figura 12 – Fluxograma das etapas de pré-processamento, treinamento e avaliação do modelo.	26
Figura 13 – Exemplo de imagem rotulada.	27
Figura 14 – Imagens de um otólito pré-processado utilizando o <i>rembg</i>	28
Figura 15 – Distribuição de frequência das idades dos peixes.	31
Figura 16 – Distribuições de frequência das idades dos peixes por ano de coleta.	32
Figura 17 – Curvas de aprendizado.	33
Figura A1 – Dispersão dos valores observados <i>versus</i> valores preditos sem <i>Test-Time Augmentation</i>	41
Figura A2 – Dispersão dos valores observados <i>versus</i> valores preditos com <i>Test-Time Augmentation</i>	42
Figura A3 – Dispersão dos resíduos <i>versus</i> valores preditos sem <i>Test-Time Augmentation</i>	43
Figura A4 – Dispersão dos resíduos <i>versus</i> valores preditos com <i>Test-Time Augmentation</i>	44
Figura B1 – Imagens de otólitos coletados em diferentes anos.	45
Figura B2 – Exemplo de limitação na remoção de fundo.	46

LISTA DE TABELAS

Tabela 1 – Tabela verdade do operador "e".	14
Tabela 2 – Medidas descritivas da idade dos peixes por ano de coleta.	32
Tabela 3 – Desempenho dos modelos sem <i>Test-Time Augmentation</i>	34
Tabela 4 – Desempenho dos modelos com <i>Test-Time Augmentation</i>	34

LISTA DE ABREVIATURAS E SIGLAS

Adam *Adaptive Moment Estimation*

CV Coeficiente de variação

EQM Erro quadrático médio

IA Inteligência artificial

IMR *Institute of Marine Research*

ReLU *Rectified Linear Unit*

RNA Rede neural artificial

RNC Rede neural convolucional

TTA *Test-Time Augmentation*

YOLO *You Only Look Once*

SUMÁRIO

1	Introdução	11
2	Fundamentação teórica	13
2.1	Modelo de um neurônio	14
2.2	Função de ativação	16
2.3	Redes <i>feedforward</i> com camada única	17
2.4	Redes <i>feedforward</i> com múltiplas camadas	18
2.5	Redes convolucionais	19
2.6	Arquitetura InceptionV3	20
2.7	rembg	22
2.8	YOLO	23
3	Metodologia	25
3.1	Detalhes da base de dados	25
3.2	Pré-processamento dos dados	26
3.2.1	Detecção e recorte automático	26
3.2.2	Extração de região de interesse	27
3.3	Treinamento e avaliação do modelo	28
3.3.1	Estratégias de Treinamento Avaliadas	30
4	Resultados	31
4.1	Estatística descritiva	31
4.2	Curvas de aprendizagem	33
4.3	Avaliação dos modelos	34
5	Discussão	36
	Referências	39
	APÊNDICE A Resultados complementares	41
	APÊNDICE B Exemplos específicos	45

INTRODUÇÃO

No âmbito da ecologia pesqueira, compreender a estrutura etária das populações de peixes é fundamental para garantir seu uso sustentável e conservação a longo prazo (CAMPANA, 2001; CARBONARA et al., 2019). Os dados de idade são importantes para analisar padrões de recrutamento, crescimento, maturidade e taxas de mortalidade dos peixes. Atualmente o método mais recomendável para a determinação da idade de peixes é a análise de estruturas rígidas, como os otólitos, que são estruturas calcificadas presentes no ouvido interno dos peixes que apresentam anéis de crescimento semelhantes aos de árvores (WANG; WALTHER; GILLANDERS, 2019). Essas estruturas são preparadas e analisadas em laboratório para contagem dos anéis de crescimento por especialistas humanos. No entanto, determinar a idade de forma manual é demorado, caro e requer experiência em leitura de otólitos. Além disso, a interpretação e contagem dos anéis de crescimento pode variar mesmo entre especialistas experientes.

Isso tem motivado o desenvolvimento de alternativas automáticas e economicamente viáveis, baseadas em técnicas de visão computacional e aprendizado de máquina. Diversos autores têm explorado o uso de modelos de redes neurais convolucionais (RNC) para estimar a idade de peixes por meio da análise de imagens de otólitos (MOEN et al., 2018; POLITIKOS et al., 2021; SIGURÐARDÓTTIR et al., 2023; İŞGÜZAR et al., 2024).

Especificamente, Moen et al. (2018) ajustaram um modelo de RNC para estimar a idade de peixes a partir de imagens de otólitos do halibute da Groenlândia (*Reinhardtius hippoglossoides*). No estudo, a idade foi corretamente predita em 48 dos 164 pares de otólitos testados (29%) e, em 63 casos (38%), a estimativa apresentou diferença de apenas um ano em relação à idade lida pelo especialista. Contudo, os autores destacam a necessidade de melhorias na qualidade das imagens de treinamento e no balanceamento dos dados para aumentar a precisão do modelo. Neste sentido, hipotetizamos que a aplicação de técnicas de aprimoramento de imagem nestes otólitos pode contribuir para um modelo mais preciso com melhor capacidade de prever a idade dos peixes.

Ma et al. (2024) reforçam essa hipótese ao aplicar técnicas de aprimoramento de ima-

gem em otólitos de atum-do-pacífico (*Thunnus orientalis*) resultando em um melhor desempenho em relação ao modelo com as imagens brutas, sem tratamento. De forma semelhante, no contexto da radiologia médica, Michael e Yoon (2020) demonstraram que a remoção de ruídos em radiografias de tórax melhora o desempenho de modelos pré-treinados, evidenciando o potencial das técnicas de pré-processamento de imagem para aprimorar tarefas de classificação.

Neste sentido, o objetivo geral deste trabalho foi avaliar estratégias de aprimoramento do processo de predição de idade a partir de imagens de otólitos de halibute da Groenlândia (*Reinhardtius hippoglossoides*), por meio da aplicação de técnicas de pré-processamento, modificações no procedimento de treinamento e estratégias de pós-processamento, com o intuito de verificar se tais intervenções contribuem para uma melhoria significativa no desempenho preditivo de um modelo de RNC. Para isso, utilizamos as imagens disponibilizadas por Moen et al. (2018), bem como o modelo de RNC originalmente proposto pelos autores, sobre o qual realizamos novos treinamentos e ajustes metodológicos. Como objetivos específicos, destacam-se:

- i) Desenvolver e implementar um protocolo automatizado de pré-processamento para remoção de fundo e padronização das imagens dos otólitos;
- ii) Investigar modificações no processo de treinamento, incluindo estratégias de amostragem estratificada, técnicas de balanceamento de dados e ajustes de hiperparâmetros;
- iii) Avaliar o impacto de estratégias de pós-processamento nas métricas de desempenho do modelo.

FUNDAMENTAÇÃO TEÓRICA

"Uma rede neural artificial (RNA) é uma máquina projetada para modelar como o cérebro realiza uma tarefa. Ela é um processador altamente distribuído em paralelo constituído de unidades de processamento simples, com uma propensão natural para armazenar conhecimento empírico e torná-lo disponível para uso" (HAYKIN, 1994, p. 24).

As unidades de processamento são chamadas "neurônios" e estão interligadas através de "sinapses", cujos pesos sinápticos também carregam informação. O processo de aprendizagem se dá pelo algoritmo de aprendizagem, cuja função é modificar os pesos sinápticos da rede de forma ordenada para atingir um objetivo de projeto desejado. Para isso, utilizamos de métodos de otimização, como a retropropagação, o método do gradiente estocástico ou o método de Newton (FRIEDLANDER, 1994).

Atualmente, as redes neurais artificiais têm sido amplamente aplicadas em diversas áreas, como:

- Visão computacional: reconhecimento facial, detecção de objetos e identificação de imagens.
- Processamento de linguagem natural: tradução automática, assistentes virtuais e transcrição de texto.
- Ciência de dados e previsão: previsão de séries temporais, detecção de fraudes e análise de mercado.
- Saúde: Processamento e análise de imagens médicas, e diagnóstico médico assistido por inteligência artificial (IA).
- Automação e controle: carros autônomos, robótica e sistemas de recomendação.

Tais aplicações evidenciam a capacidade das RNAs de resolver problemas em diferentes áreas, reforçando sua relevância como uma tecnologia fundamental na era da inteligência artificial.

2.1 MODELO DE UM NEURÔNIO

Um neurônio é uma unidade de processamento de informação que é fundamental para a operação de uma RNA (conferir a representação do neurônio k na Figura 1). Nele, temos três elementos básicos:

- Um conjunto de sinapses $w_{k1}, w_{k2}, \dots, w_{kn}$, caracterizados por seu peso sináptico. Por exemplo, a entrada x_j está conectada ao neurônio k pela sinapse w_{kj} .
- Um somador para somar os sinais da camada anterior, ponderados pelos pesos sinápticos.
- Uma função de ativação φ para restringir a amplitude de saída de um neurônio.

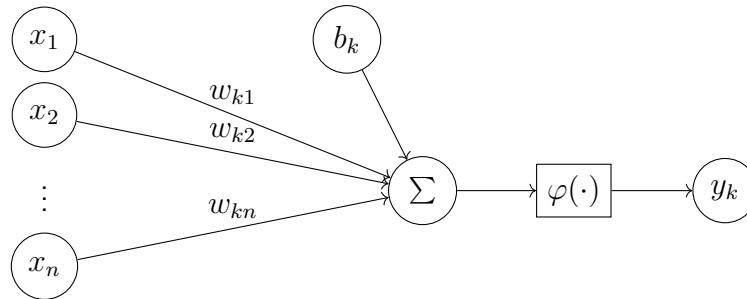


Figura 1 – Modelo de um neurônio k .

Além disso, temos na Figura 1 os sinais de entrada $x_1, x_2, \dots, x_n$, o sinal de saída y_k e o *bias* (ou viés) b_k . Podemos descrever um neurônio k de forma matemática com as seguintes equações:

$$u_k = \sum_{j=1}^n w_{kj} x_j,$$

e

$$y_k = \varphi(u_k + b_k).$$

Usamos o *bias* para aplicar uma transformação afim (translação, rotação, etc.) na combinação linear u_k . A seguir, veremos um exemplo do funcionamento de uma RNA.

Exemplo: Vamos construir uma RNA que modele a relação de entrada e saída descrito pela estrutura lógica "e" (Tabela 1).

Tabela 1 – Tabela verdade do operador "e".

x_1	x_2	x_1 e x_2
1	1	1
1	0	0
0	1	0
0	0	0

Podemos ver na Figura 2 uma RNA, com função de ativação (2.1), que após um processo de otimização, tem os valores dos pesos sinápticos $w_{11} = w_{12} = 1$ e o valor do *bias* $b_1 = -2$.

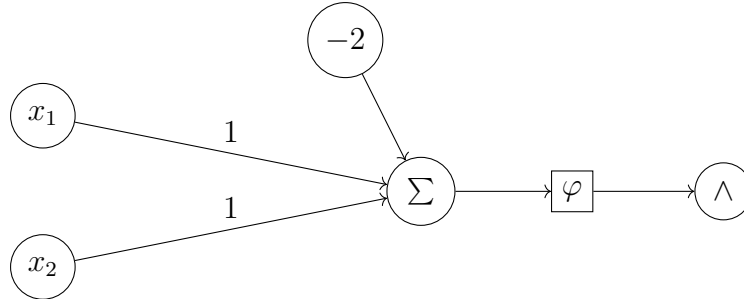


Figura 2 – Exemplo de RNA.

$$\varphi(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases}.$$

Desta forma, vamos conferir seu funcionamento. Para $x_1 = 1$ e $x_2 = 1$ temos

$$\begin{aligned} u_1 &= \sum_{j=1}^2 w_{1j}x_j \implies \\ u_1 &= (w_{11} \times x_1) + (w_{12} \times x_2) \implies \\ u_1 &= (1 \times 1) + (1 \times 1) \implies \\ u_1 &= 2 \implies \\ y &= \varphi(u_1 + b_1) \implies \\ y &= \varphi(2 + (-2)) \\ y &= 0. \end{aligned}$$

Para $x_1 = 1$ e $x_2 = 0$ temos

$$\begin{aligned} u_1 &= (1 \times 1) + (1 \times 0) \implies \\ u_1 &= 1 \implies \\ y &= \varphi(1 - 2) \implies \\ y &= \varphi(-1) \implies \\ y &= 0. \end{aligned}$$

Para $x_1 = 0$ e $x_2 = 1$ temos

$$u_1 = (1 \times 0) + (1 \times 1) \implies$$

$$u_1 = 1 \implies$$

$$y = \varphi(1 - 2) \implies$$

$$y = \varphi(-1) \implies$$

$$y = 0.$$

Para $x_1 = 0$ e $x_2 = 0$ temos

$$u_1 = (1 \times 0) + (1 \times 0) \implies$$

$$u_1 = 0 \implies$$

$$y = \varphi(0 - 2) \implies$$

$$y = \varphi(-2) \implies$$

$$y = 0.$$

Logo, a RNA consegue descrever a Tabela 1, conforme esperado.

2.2 FUNÇÃO DE ATIVAÇÃO

Várias funções de ativação são mostradas na Figura 3, note que o traço destas funções tem o formato de "S". Porém podemos utilizar outras funções, a função Gaussiana, por exemplo

$$\varphi(x) = e^{-x^2},$$

que é usada como ativação principalmente em redes voltadas para aproximação de funções.

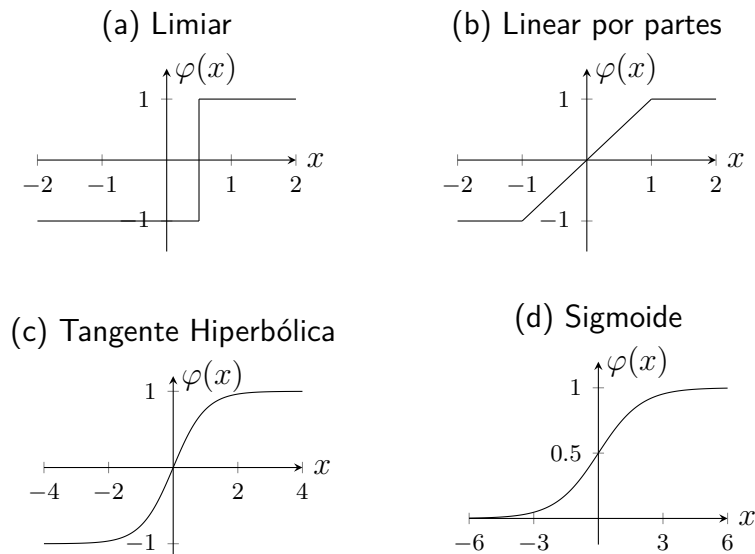


Figura 3 – Gráficos de funções de ativação.

As funções de ativação tem como objetivo introduzir não linearidade no modelo e limitar o domínio de entrada infinito para um intervalo de saída finito, também são chamadas de função de esmagamento (*squashing function*). A função limiar (ou *threshold*), ilustrada na Figura 3a é dada por

$$\varphi(x) = \begin{cases} 1, & \text{se } x \geq \frac{1}{2} \\ 0, & \text{se } x < \frac{1}{2} \end{cases}.$$

Na Figura 3b temos o gráfico da função linear por partes, dada por

$$\varphi(x) = \begin{cases} 1, & \text{se } x \geq 1 \\ X, & \text{se } -1 < x < 1 \\ -1, & \text{se } x \leq -1 \end{cases}.$$

As Figuras 3c e 3d ilustram os gráficos das funções tangente hiperbólica e sigmoide, respectivamente, dadas por

$$\varphi(x) = \tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}, \quad (2.1)$$

e

$$\varphi(x) = \sigma(x) = \frac{1}{1 + e^{-x}}. \quad (2.2)$$

De acordo com Graves (2012), (2.1) e (2.2) são equivalentes como funções de ativação. Entretanto, uma distinção relevante entre elas está nos intervalos de saída que produzem. Em particular, quando se deseja uma saída no intervalo entre 0 e 1, como no caso em que a saída representa uma probabilidade, a função sigmoide é a mais adequada. Uma característica importante tanto da tangente hiperbólica quanto da sigmoide é a sua não linearidade. Redes neurais com funções de ativação não lineares são mais poderosas em comparação às redes com funções de ativação lineares, pois conseguem, por exemplo, identificar fronteiras de decisão e modelar relações não lineares.

2.3 REDES *FEEDFORWARD* COM CAMADA ÚNICA

Uma rede de neurônios é organizada em camadas, na forma mais simples, temos uma camada de entrada (com os dados) e uma camada de saída (com os neurônios que fazem os cálculos). Os dados fluem apenas da entrada para a saída, sem retorno, isso é chamado de rede *feedforward* ou acíclica.

Esse tipo de rede, ilustrada na Figura 4, quando tem apenas uma camada de neurônios (a de saída), é chamada de rede de camada única. A camada de entrada não é contada porque só fornece os dados e não realiza cálculos. Na próxima seção veremos um tipo de RNA com múltiplas camadas de neurônios.

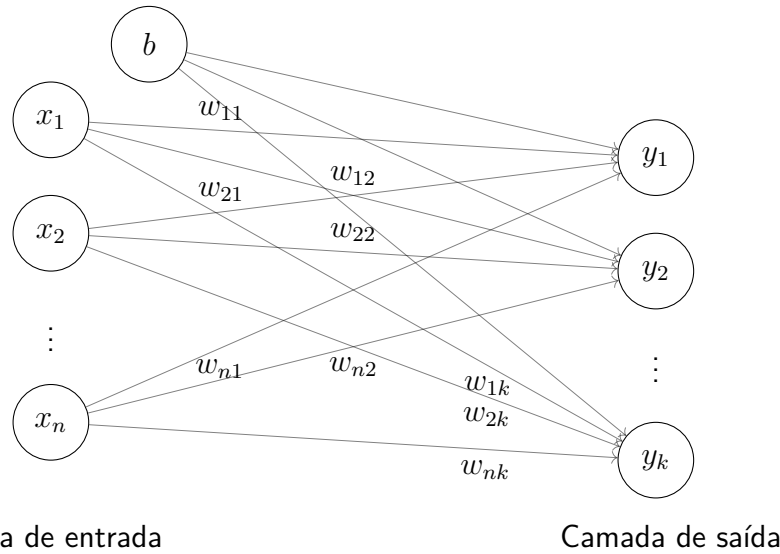


Figura 4 – Modelo de RNA com n neurônios de entrada e k neurônios de saída.

2.4 REDES *FEEDFORWARD* COM MÚLTIPLAS CAMADAS

Redes *feedforward* com múltiplas camadas são RNAs que se diferem da vista anteriormente, por apresentar uma ou mais camadas ocultas, isto é, camadas intermediárias, entre as camadas de entrada e saída. Ainda, as unidades de processamento destas camadas ocultas são denominadas de neurônios ocultos. Eles têm esse nome simplesmente por não serem neurônios de entrada e nem de saída. O papel da camada oculta na rede é extrair estatísticas de ordem superior, o que é especialmente útil quando a entrada contém muitas variáveis. Mesmo que as conexões entre os neurônios sejam locais, a rede adquire uma visão mais global do problema por conta dessas novas camadas e conexões. Isso amplia a capacidade de representação da rede — indo além das relações simples que uma rede de camada única consegue identificar.

A Figura 5 ilustra este tipo de rede, que ainda é alimentada para a frente, os dados entram pela camada de entrada e passam para a primeira camada oculta, na qual são processados. O resultado é enviado para a próxima camada, e assim por diante, até chegar na camada de saída. Cada camada normalmente recebe como entrada os sinais da camada anterior e gera sinais que servirão de entrada para a próxima.

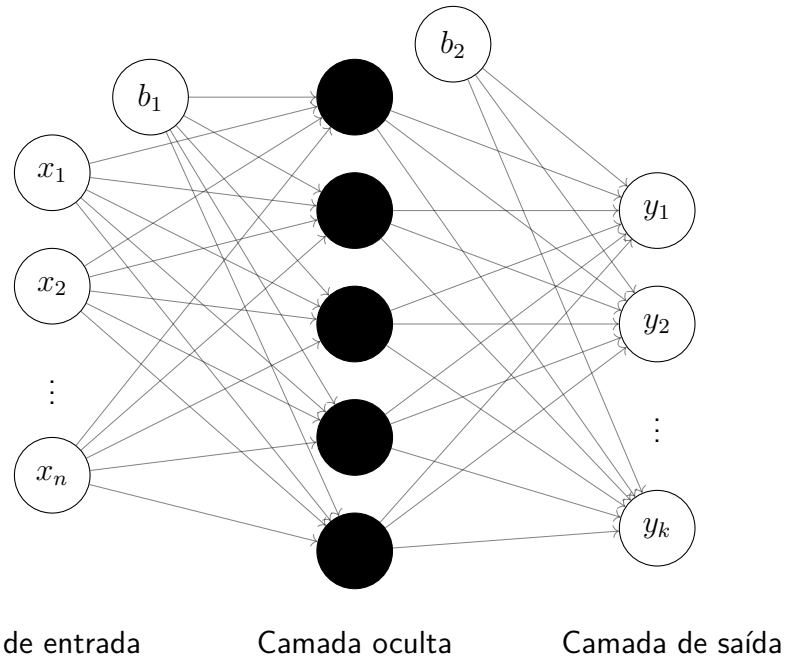


Figura 5 – Modelo de RNA com n neurônios de entrada, uma camada oculta com 5 neurônios e k neurônios de saída.

Além disso, uma rede é chamada de totalmente conectada quando todos os neurônios de uma camada estão ligados a todos os neurônios da camada seguinte. Se algumas dessas conexões estiverem ausentes, a rede é considerada parcialmente conectada. Quando temos múltiplas camadas ocultas, dizemos que a rede é uma rede neural profunda.

2.5 REDES CONVOLUCIONAIS

As RNCs são uma categoria de rede neural profunda especializada em processar dados que apresentam uma topologia em grade. Por exemplo, imagens digitais, representadas como uma grade bidimensional de pixels. As RNCs têm se mostrado extremamente eficazes em aplicações práticas, em especial na classificação de imagens.

O nome "rede neural convolucional" refere-se ao uso, por essas redes, de uma operação matemática denominada convolução. RNCs são simplesmente RNAs que usam a convolução em pelo menos uma de suas camadas (GOODFELLOW; BENGIO; COURVILLE, 2016).

A convolução é um tipo particular de operação linear. Neste contexto, é uma forma de extrair padrões (como bordas, texturas, formas) de uma imagem. Aplica-se um filtro (ou *kernel*) a uma matriz de entrada (como uma imagem). Matematicamente, a convolução discreta bidimensional entre uma imagem I e um filtro K pode ser representada por:

$$S(i, j) = \sum_m \sum_n I(m, n) K(i - m, j - n),$$

em que $S(i, j)$ é o valor resultante na posição (i, j) da imagem de saída, $I(m, n)$ representa os valores dos pixels da imagem de entrada e K é o filtro aplicado com os índices invertidos (refletidos). A Figura 6 a ilustra uma convolução sem o filtro refletido. Na prática, a convolução consiste em deslocar o filtro ao longo da imagem e construir uma nova imagem chamada mapa de ativação ou *feature map*, que destaca determinadas características da imagem original. Em conjunto com a convolução temos a operação de subamostragem ou *pooling*, que reduz as dimensões espaciais dos mapas de ativação, ao mesmo tempo que retém as informações mais importantes. Desta forma, a convolução contribui para a eficiência computacional da RNA ao reduzir o número de parâmetros e explorar a estrutura local dos dados.

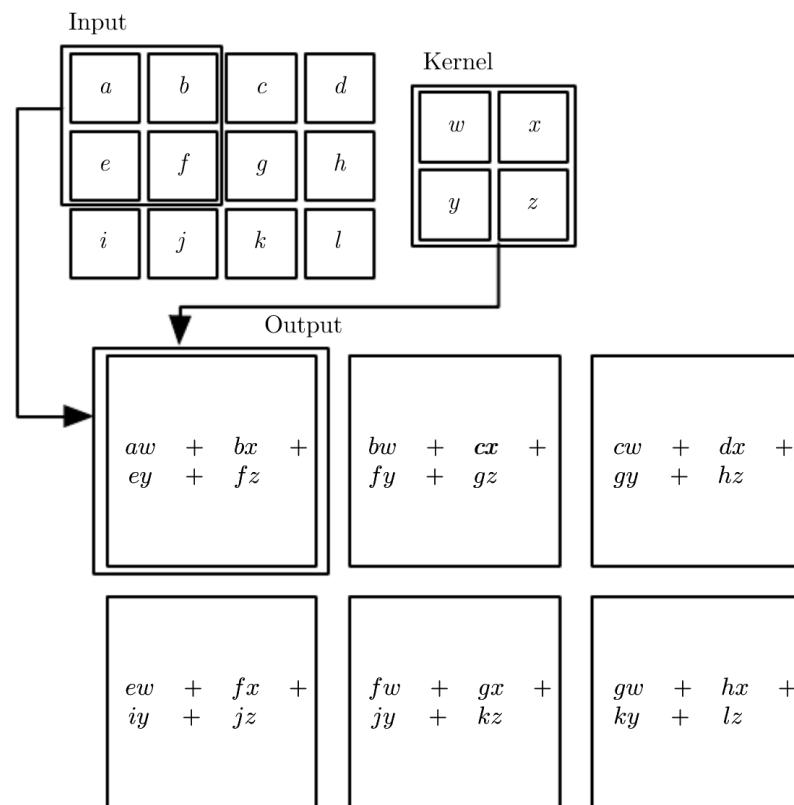


Figura 6 – Exemplo de convolução 2-D. Fonte: (GOODFELLOW; BENGIO; COURVILLE, 2016).

Para aprofundar os conhecimentos sobre redes neurais artificiais e redes neurais convolucionais, recomenda-se a consulta das referências, que oferecem uma base teórica sólida e exemplos práticos sobre os temas abordados (HAYKIN, 1994; GOODFELLOW; BENGIO; COURVILLE, 2016).

2.6 ARQUITETURA INCEPTIONV3

A InceptionV3 é uma arquitetura de RNC para classificação de imagens com 48 camadas de neurônios (SZEGEDY et al., 2015). Essa arquitetura é uma evolução das versões

anteriores da família *Inception*, com melhorias que a tornam mais eficiente em termos de desempenho e custo computacional. Um componente central da arquitetura são os módulos *Inception*, que realizam múltiplas operações em paralelo dentro de um mesmo bloco, como convoluções de diferentes tamanhos (por exemplo, 1×1 , 3×3) e *pooling* (Figura 7).

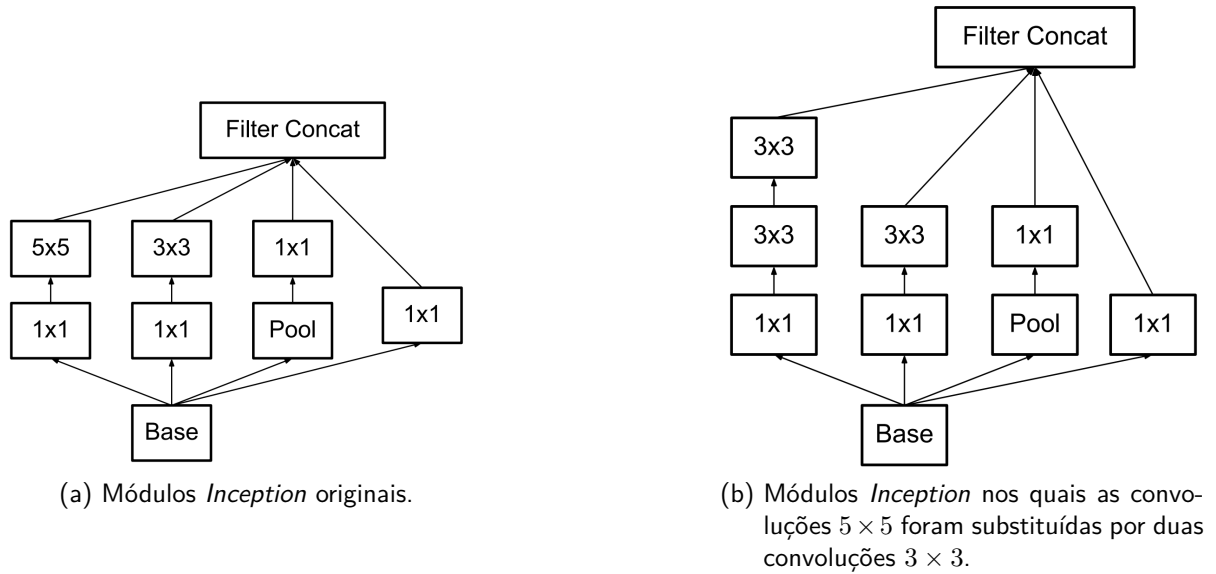


Figura 7 – Exemplos de módulo *Inception*. Fonte: (SZEGEDY et al., 2015).

Uma das inovações mais importantes introduzidas na InceptionV3 é o uso de convoluções fatoradas. Por exemplo, em vez de aplicar diretamente uma convolução 5×5 , que é mais custosa, o modelo aplica duas convoluções 3×3 em sequência, o que reduz o número de parâmetros e acelera o treinamento (Figuras 7a e 7b). A InceptionV3 utiliza majoritariamente a função de ativação ReLU (*Rectified Linear Unit*) (2.3) em suas camadas convolucionais e totalmente conectadas, já na camada de saída utiliza a função de ativação *softmax* (2.4) para tarefas de classificação. Embora originalmente projetada para tarefas de classificação, a arquitetura pode ser adaptada para diversos tipos de problemas, como regressão, apenas ajustando as camadas finais e trocando a função de ativação para a função linear (2.5).

$$\varphi(x) = R(x) = \max(0, x), \quad (2.3)$$

$$\varphi(x) = s(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \quad (2.4)$$

$$\varphi(x) = x. \quad (2.5)$$

A Figura 8 ilustra a arquitetura da rede InceptionV3, da esquerda para a direita. O modelo recebe imagens no formato $299 \times 299 \times 3$ (altura, largura e canais RGB), os módulos *Inception* combinam convoluções de diferentes tamanhos de filtro e operações de pooling em paralelo, concatenando os resultados. Entre grupos de módulos *Inception*, são aplicadas etapas de redução de dimensionalidade (*grid size reduction*), que reduzem as dimensões espaciais do

mapa de ativação, ao mesmo tempo em que aumentam a profundidade (número de filtros), tornando o modelo mais eficiente. Além disso, durante o treinamento, é utilizado um classificador auxiliar (*auxiliary classifier*), que atua como forma de regularização e ajuda a mitigar o problema do desaparecimento do gradiente (*vanishing gradient problem*). Assim, a saída intermediária do modelo é um tensor de dimensão $8 \times 8 \times 2048$, que, após as camadas finais, resulta em uma predição de probabilidade para cada classe (no caso original, 1001 classes do ImageNet).

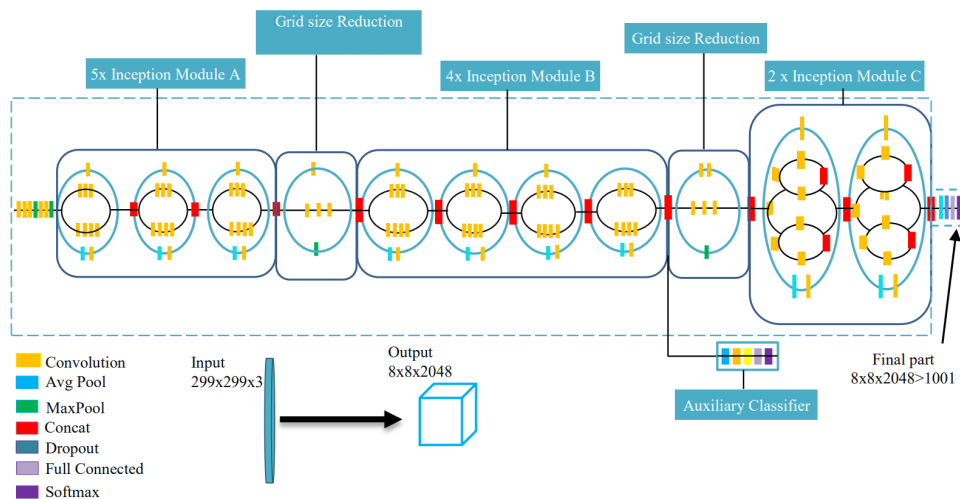


Figura 8 – Arquitetura InceptionV3. Fonte: (IPARRAGUIRRE-VILLANUEVA et al., 2022)

Devido à sua robustez e generalização, a InceptionV3 é frequentemente utilizada em abordagens de aprendizado por transferência (*transfer learning*), nas quais pesos previamente treinados em grandes conjuntos de dados, como o ImageNet, são reutilizados para treinamentos em outro conjunto de dados.

2.7 REMBG

A remoção de fundo é uma etapa relevante no pré-processamento de imagens quando o objetivo é isolar um objeto de interesse, reduzindo a influência de informações irrelevantes no processo de análise da imagem. Em problemas de visão computacional, especialmente aqueles baseados em *deep learning*, a presença de fundos complexos pode introduzir ruído, dificultar o aprendizado de características discriminantes e impactar negativamente o desempenho dos modelos.

O *rembg* é uma ferramenta de código aberto destinada à remoção automática de fundo em imagens digitais, baseada em RNCs treinadas para tarefas de segmentação semântica (GATIS, 2025). Seu funcionamento consiste em classificar cada pixel da imagem como pertencente ao objeto principal (*foreground*) ou ao fundo (*background*), produzindo como saída uma máscara binária que permite separar essas duas regiões.

Internamente, o *rembg* utiliza modelos da arquitetura U²-Net, uma RNC desenvolvida para a segmentação de objetos salientes em imagens (QIN et al., 2020). Caracterizada por uma estrutura em forma de "U dentro de U", que combina múltiplos níveis de resolução espacial a fim de capturar simultaneamente informações globais e detalhes locais da imagem, sendo particularmente adequada para objetos com contornos irregulares ou variações sutis de textura. A partir da segmentação, é gerada uma máscara que classifica os pixels da imagem como pertencentes ao objeto de interesse ou ao fundo, a qual é posteriormente aplicada à imagem original para remover a região de fundo e preservar apenas o objeto segmentado (Figura 9).



(a) Imagem original.



(b) Imagem com o fundo removido.

Figura 9 – Exemplo de imagem com o fundo removido pelo *rembg*. Fonte: (GATIS, 2025).

Neste contexto, a utilização do *rembg* como etapa de pré-processamento pode contribuir para a melhoria do desempenho dos modelos de visão computacional, ao direcionar a atenção da rede neural exclusivamente para as características morfológicas do otólito. Além disso, a automatização do processo permite o tratamento de grandes volumes de dados de forma reprodutível, reduzindo a subjetividade inerente a métodos manuais de segmentação de imagem.

2.8 YOLO

A detecção de objetos consiste na tarefa de identificar e localizar automaticamente instâncias de objetos de interesse em imagens. Diferentemente da classificação de imagens, na qual se atribui um único rótulo à imagem inteira, a detecção de objetos busca determinar tanto a classe quanto a posição espacial de cada objeto presente, geralmente por meio de caixas delimitadoras (*bounding boxes*) (Figura 10).

Entre as arquiteturas desenvolvidas para essa finalidade, destaca-se o YOLO (*You Only Look Once*), proposto por Redmon et al. (2015). O YOLO é um modelo de RNC projetado para realizar detecção de objetos em tempo real, reformulando o problema como uma única

tarefa de regressão direta, que mapeia a imagem de entrada para as coordenadas das caixas delimitadoras e suas respectivas probabilidades de classe.

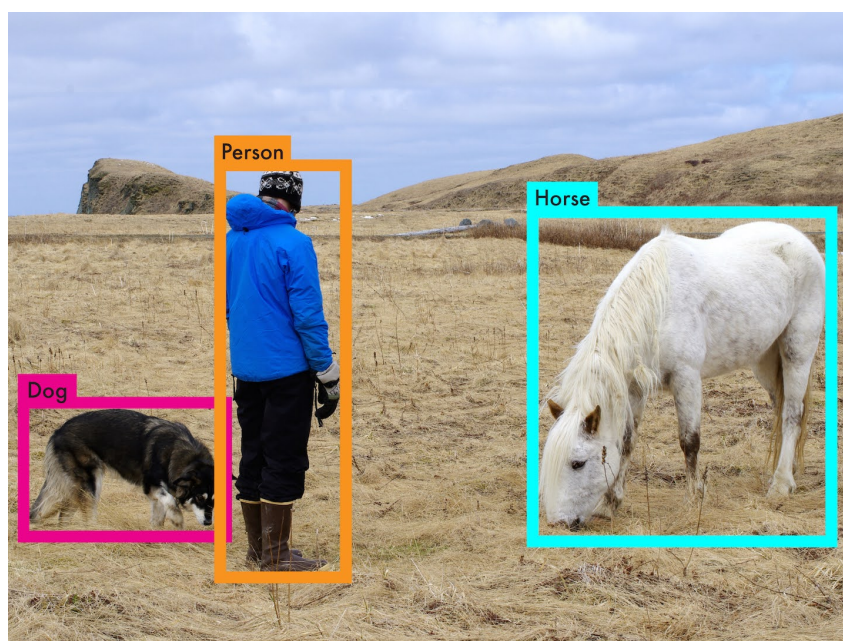


Figura 10 – Exemplo de detecção de objetos utilizando YOLO. Fonte: (REDMON et al., 2015).

Diferentemente de abordagens baseadas em múltiplas etapas, o YOLO processa a imagem inteira em uma única passagem pela rede. De forma geral, a arquitetura é composta por uma rede convolucional profunda responsável pela extração de características, seguida por camadas que realizam a regressão das coordenadas das caixas delimitadoras e a estimativa das probabilidades de detecção.

METODOLOGIA

3.1 DETALHES DA BASE DE DADOS

O conjunto de dados utilizado neste estudo foi composto por imagens de otólitos, provenientes do acervo de imagens do *Institute of Marine Research* (IMR), localizado em Bergen, Noruega. Os otólitos foram coletados e fotografados como parte do programa de monitoramento do IMR durante cruzeiros de pesquisa realizados entre os anos de 2006 e 2017, com foco no halibute da Groenlândia (*Reinhardtius hippoglossoides*) (Figura 11).



Figura 11 – Coleta do otólito do halibute. Fonte: Greenland Institute of Natural Resources.

Optou-se por utilizar apenas essa espécie porque os dados encontram-se publicamente disponíveis. O conjunto original é composto por 4109 imagens contendo pares de otólitos e 657 imagens de otólitos únicos, totalizando 8875 otólitos, removidos de 4766 peixes. A partir desses dados, Moen et al. (2018) disponibilizaram uma versão processada do conjunto, composta por 8875 imagens individuais já rotuladas, em que cada imagem contém um único otólito. Todas as imagens originais possuem dimensão de 2592×1944 pixels; no caso dos pares de otólitos, cada imagem foi dividida em duas, resultando em variação na largura das imagens individuais.

3.2 PRÉ-PROCESSAMENTO DOS DADOS

A partir do conjunto de dados original, foram definidas três estratégias distintas de tratamento das imagens: utilização das imagens sem aplicação de pré-processamento; aplicação de recorte; e aplicação de remoção de fundo, seguida de padronização de resolução e posição. Cada uma dessas estratégias resultou em um conjunto de dados específico, os quais foram utilizados como entrada para o treinamento e avaliação dos modelos. As etapas de pré-processamento associadas a cada estratégia podem ser visualizadas no fluxograma apresentado na Figura 12.

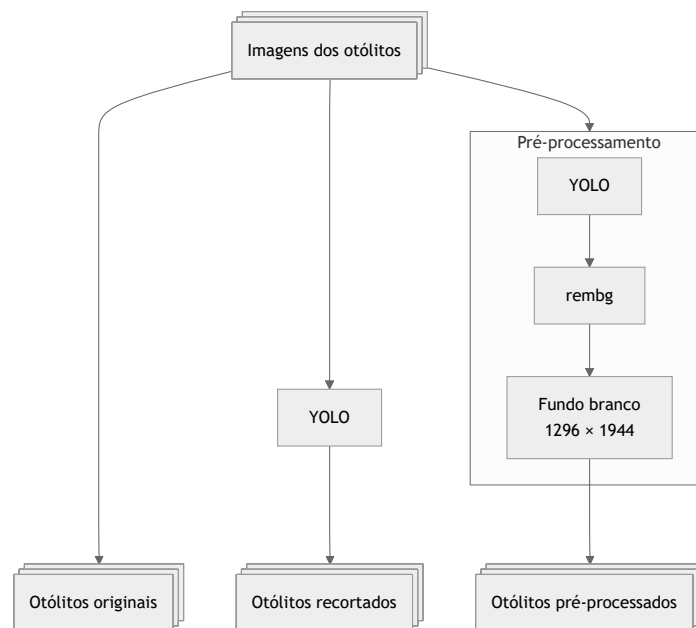


Figura 12 – Fluxograma das etapas de pré-processamento, treinamento e avaliação do modelo.

3.2.1 DETECÇÃO E RECORTE AUTOMÁTICO

O primeiro pré-processamento de imagem que realizamos foi o recorte das imagens de forma a conter apenas o otólito. Para isso, uma amostra aleatória de 500 imagens foi rotulada manualmente, identificando a região que corresponde ao otólito (Figura 13). A rotulagem foi realizada por meio da ferramenta web Roboflow. As imagens foram divididas de forma aleatória em dois subconjuntos: 400 para o treinamento e 100 para a validação. Com base nesse conjunto, foi treinado um modelo YOLO, utilizando a linguagem de programação Python. Após o treinamento, o modelo foi utilizado para detectar automaticamente os otólitos nas imagens, permitindo o recorte das regiões identificadas. O modelo foi capaz de identificar otólitos em todas as imagens avaliadas, e o recorte foi implementado diretamente no script de detecção. As imagens utilizadas no próprio processo de treinamento e validação também foram recortadas com base nas anotações originais.

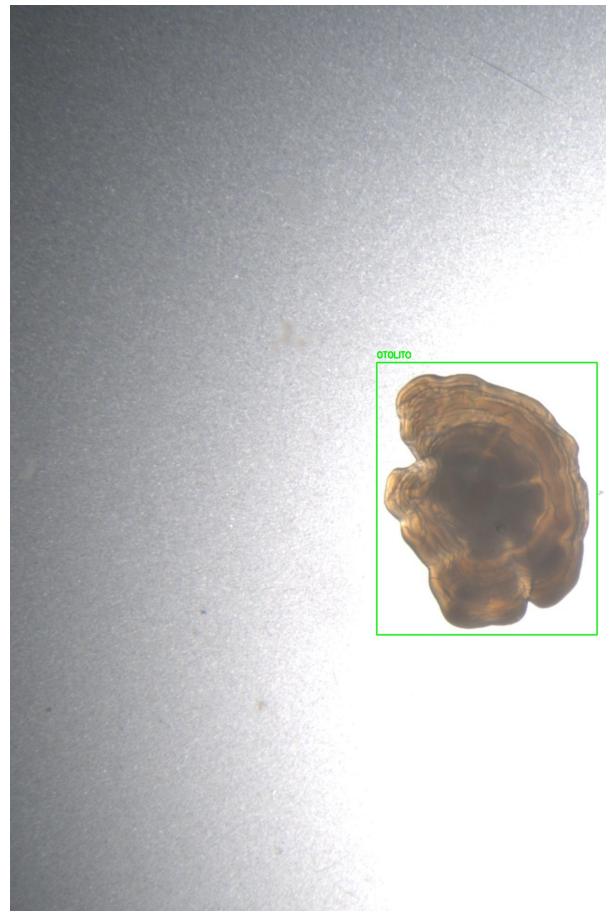


Figura 13 – Exemplo de imagem rotulada.

3.2.2 EXTRAÇÃO DE REGIÃO DE INTERESSE

Após o recorte automático dos otólitos, foi realizada outra etapa de pré-processamento voltada à remoção do fundo das imagens. Essa etapa teve como objetivo isolar a estrutura de interesse e reduzir a influência de elementos irrelevantes ao processo de aprendizagem do modelo, como ruídos de fundo e artefatos oriundos do processo de aquisição das imagens. Para isso, foi utilizada a ferramenta *rembg* como biblioteca da linguagem Python. A partir das imagens recortadas, foi removido o fundo e o otólito foi inserido em uma imagem com fundo branco e resolução fixa de 1296×1944 pixels, com posicionamento no centro da imagem, padronizando o tamanho e a resolução das imagens (Figura 14).

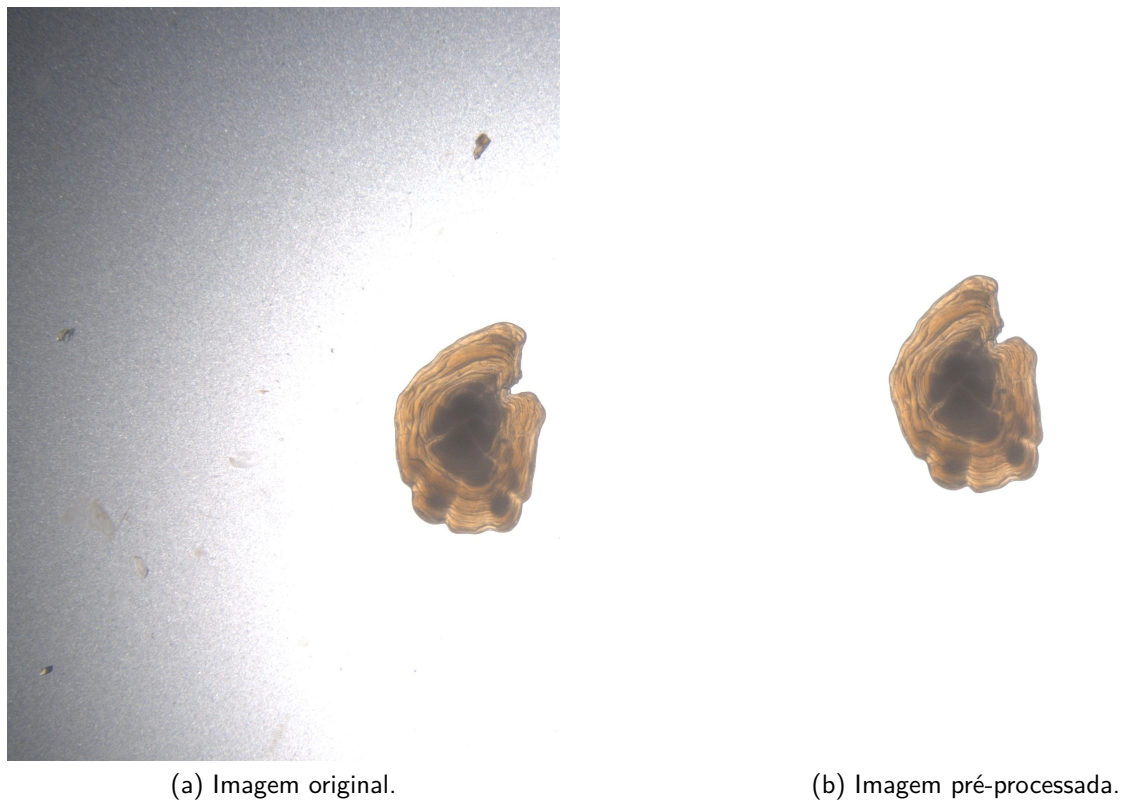


Figura 14 – Imagens de um otólito pré-processado utilizando o *rembg*. Note que todo o ruído de fundo da imagem (a) é removido na imagem (b), isolando o otólito na imagem. Além disso, os otólitos sempre ocuparão a mesma posição na imagem.

3.3 TREINAMENTO E AVALIAÇÃO DO MODELO

Um modelo de RNC foi treinado utilizando imagens de otólitos com o objetivo de prever a idade do indivíduo. Para isso, utilizou-se o modelo InceptionV3 da biblioteca Keras, rodando sobre o *framework* TensorFlow, implementado na linguagem de programação Python. As imagens foram redimensionadas para 400×400 pixels, sendo essa dimensão adotada como entrada da rede. A arquitetura foi configurada para regressão linear por meio da substituição da camada final de classificação por uma camada totalmente conectada com ativação linear.

O treinamento foi conduzido em duas etapas. Inicialmente, realizou-se uma fase de *warm-up* (aquecimento), durante 5 épocas, na qual as camadas convolucionais iniciais foram mantidas congeladas, permitindo o ajuste apenas das camadas finais adicionadas. Em seguida, todas as camadas foram descongeladas para o procedimento de *fine-tuning* (ajuste fino) completo da rede.

Foram utilizados os seguintes hiperparâmetros: otimizador Adam (*Adaptive Moment Estimation*), taxa de aprendizado inicial de 10^{-3} na fase de aquecimento e 10^{-5} na fase de ajuste fino, até 60 épocas de treinamento e *batch size* (tamanho de lote) 36. Para evitar sobreajuste, empregou-se *Early Stopping* com paciência igual a 21 épocas e restauração dos

melhores pesos, além de redução adaptativa da taxa de aprendizado com fator 10^{-1} , aplicada após 5 épocas consecutivas sem melhora na métrica de validação. Durante o treinamento, aplicou-se aumento de dados (*data augmentation*), incluindo rotações aleatórias de até 360° e espelhamento horizontal. O treinamento computacional foi realizado com uma placa de vídeo NVIDIA® GeForce RTX™ 4070 Ti, possibilitando menor tempo de processamento.

O particionamento dos dados foi realizado de forma pareada e estratificada. Inicialmente, as imagens foram agrupadas por indivíduo, garantindo que os otólitos direito e esquerdo de um mesmo peixe fossem alocados conjuntamente no mesmo subconjunto. A estratificação foi realizada com base em faixas etárias de largura três anos, assegurando representatividade da distribuição etária nos conjuntos de treinamento (70%), validação (15%) e teste (15%).

Na etapa de avaliação, foi empregada a técnica de *Test-Time Augmentation* (TTA), na qual múltiplas rotações da mesma imagem foram submetidas ao modelo. A predição final da idade foi obtida pela média das predições. A idade dos indivíduos foi calculada como a média das idades estimadas de ambos os otólitos. Para peixes com apenas um otólito disponível, foi considerada a idade predita pelo modelo.

Durante o treinamento, a função de perda adotada foi o erro quadrático médio (EQM) (3.1), de modo que os parâmetros da rede neural foram estimados por meio da minimização da mesma. Também foram calculados o coeficiente de determinação R^2 (3.2) e a média do coeficiente de variação (CV) (3.3).

$$\text{EQM} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (3.1)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad (3.2)$$

$$\text{CV}_j = \frac{\sqrt{\sum_{i=1}^K (x_{ij} - \bar{x}_j)^2 / (K - 1)}}{\bar{x}_j},$$

em que $K = 2$ é o número de estimadores, x_{ij} é a estimativa do estimador i para o otólito j , e $\bar{x}_j = \frac{1}{K} \sum_{i=1}^K x_{ij}$ é a média das estimativas para o otólito j . Para avaliar o desempenho global, o CV médio foi calculado como:

$$\overline{\text{CV}} = \frac{1}{J} \sum_{j=1}^J \text{CV}_j, \quad (3.3)$$

em que J é o número total de otólitos avaliados.

Como a idade verdadeira dos indivíduos não é conhecida, adotou-se uma abordagem comumente utilizada em estudos de validação de leitura de otólitos, nos quais se comparam diferentes leitores humanos (CAMPANA; ANNAND; MCMILLAN, 1995). Neste caso, comparamos o especialista humano com o modelo de rede neural.

3.3.1 ESTRATÉGIAS DE TREINAMENTO AVALIADAS

Com o objetivo de investigar o impacto do desbalanceamento da distribuição etária no desempenho do modelo, foram avaliadas quatro estratégias distintas de treinamento:

- **M1:** O primeiro modelo corresponde ao treinamento padrão, no qual as imagens foram amostradas segundo sua frequência natural no conjunto de dados, sem balanceamento.
- **M2:** O segundo modelo empregou uma estratégia de sobreamostragem por lote, na qual cada lote de treinamento foi construído de forma a conter proporções semelhantes de amostras provenientes de diferentes faixas etárias.
- **M3:** O terceiro modelo utilizou ponderação da função de perda, atribuindo pesos inversamente proporcionais à frequência das faixas etárias no conjunto de treinamento.
- **M4:** Por fim, o quarto modelo adotou uma estratégia híbrida, combinando sobreamostragem das faixas etárias menos frequentes com ponderação da função de perda. Para esse modelo, utilizou-se *batch size* igual a 18.

A comparação entre essas quatro estratégias permitiu avaliar o impacto de diferentes estratégias de correção do desbalanceamento sobre o desempenho preditivo do modelo.

RESULTADOS

4.1 ESTATÍSTICA DESCRITIVA

A idade dos peixes varia entre 1 e 26 anos, com mediana igual a 11 anos, média de 11,07 anos e desvio padrão de 4,38 anos. Embora a média, a mediana e a moda sejam próximas, a Figura 15 dá indícios de assimetria nos dados. É possível notar a baixa frequência de indivíduos com idade superior a 20 anos, com menos de 30 peixes por classe etária.

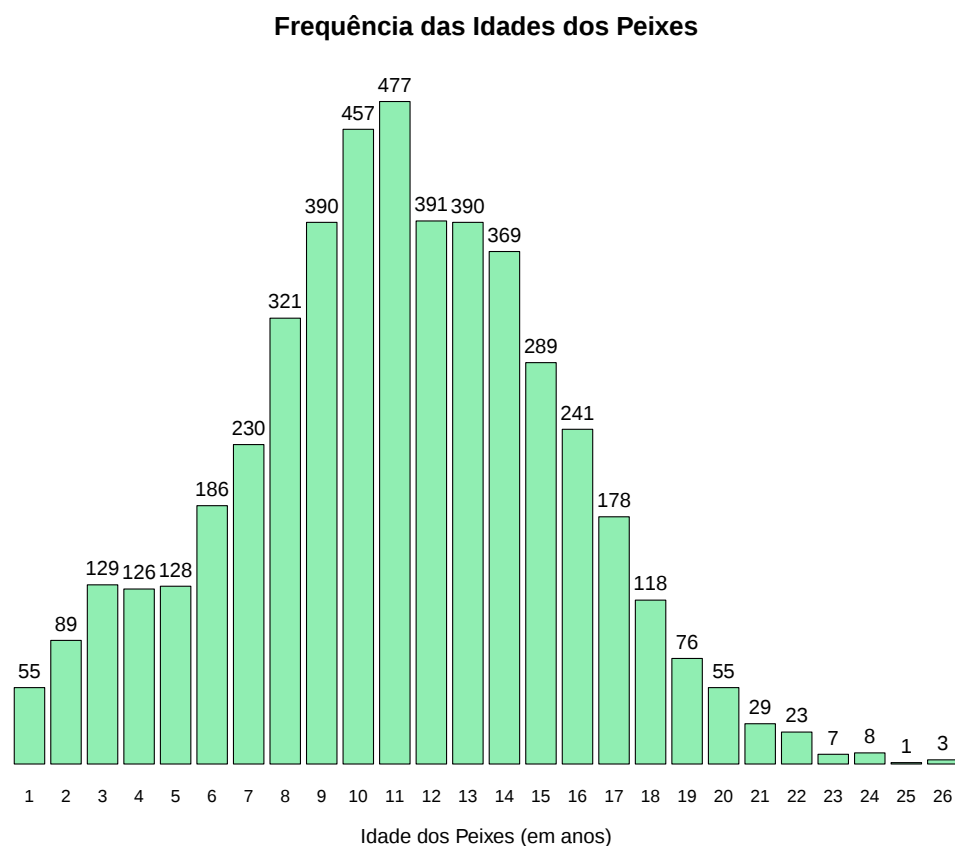


Figura 15 – Distribuição de frequência das idades dos peixes.

A distribuição das idades dos peixes ao longo dos anos de coleta apresentou indícios

de variação (Tabela 2 e Figura 16). Observa-se uma tendência de aumento nas idades médias e medianas nos anos de 2014 e 2015 em relação aos anos iniciais.

Tabela 2 – Medidas descritivas da idade dos peixes por ano de coleta.

Ano	1º Quartil	Mediana	Média	Desvio padrão	3º Quartil
2006	3,0000	4,0000	5,0679	2,8403	7,0000
2008	5,0000	7,0000	7,0736	3,0918	9,0000
2011	7,0000	9,0000	8,7133	3,6437	11,0000
2013	9,0000	11,0000	11,5267	3,7753	14,0000
2014	14,0000	16,0000	15,6194	3,0742	17,0000
2015	11,0000	13,0000	13,0993	2,9154	15,0000
2016	10,0000	12,0000	11,9070	3,5858	14,0000
2017	8,0000	11,0000	10,9785	3,5725	13,0000

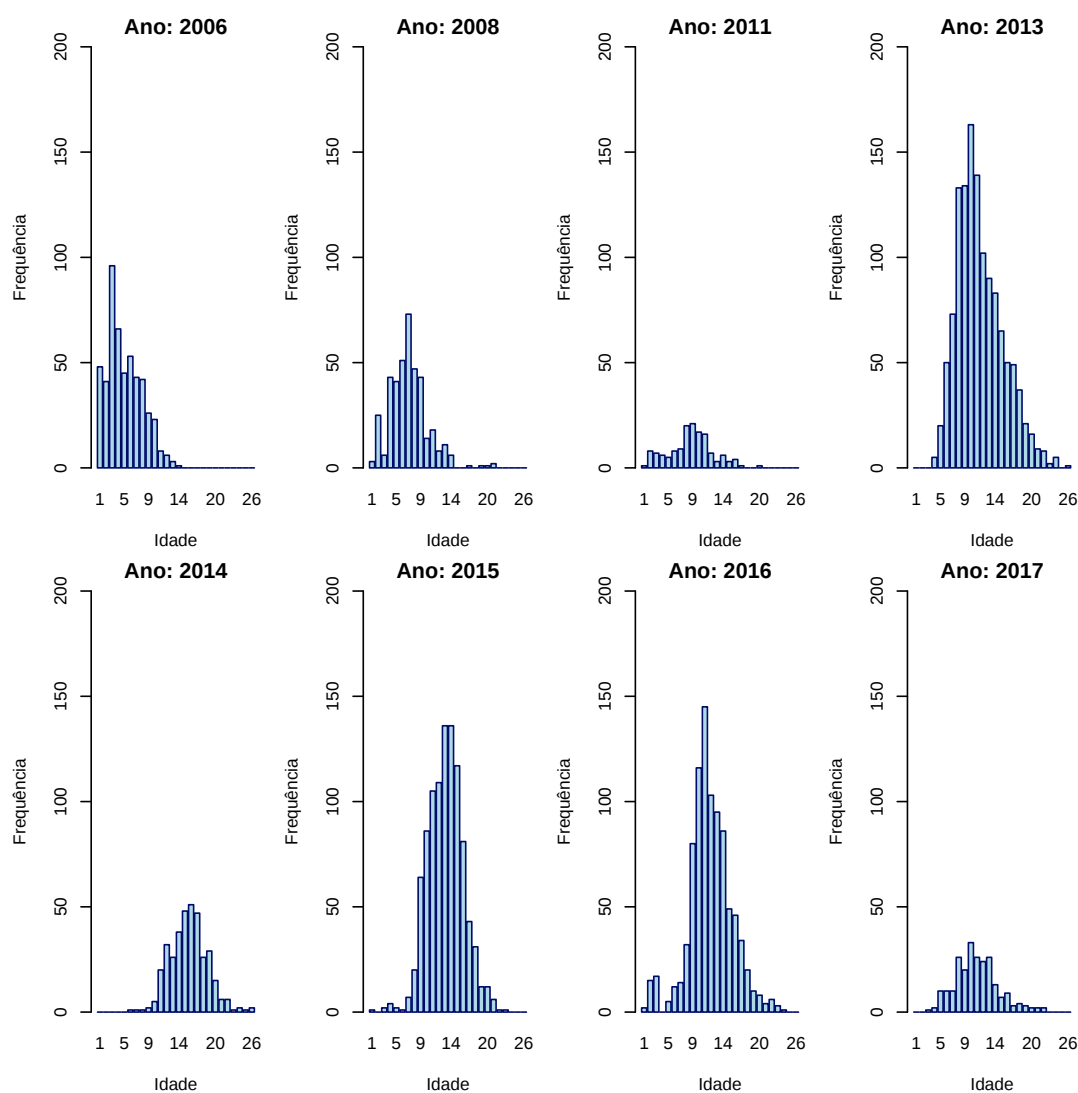


Figura 16 – Distribuições de frequência das idades dos peixes por ano de coleta.

4.2 CURVAS DE APRENDIZAGEM

A Figura 17 apresenta as curvas de aprendizagem dos modelos M1-M4, considerando os três conjuntos de imagens avaliados. Em cada painel são exibidas as trajetórias do EQM ao longo das épocas, tanto para o conjunto de treinamento quanto para o conjunto de validação, bem como os pontos de alteração da taxa de aprendizagem.

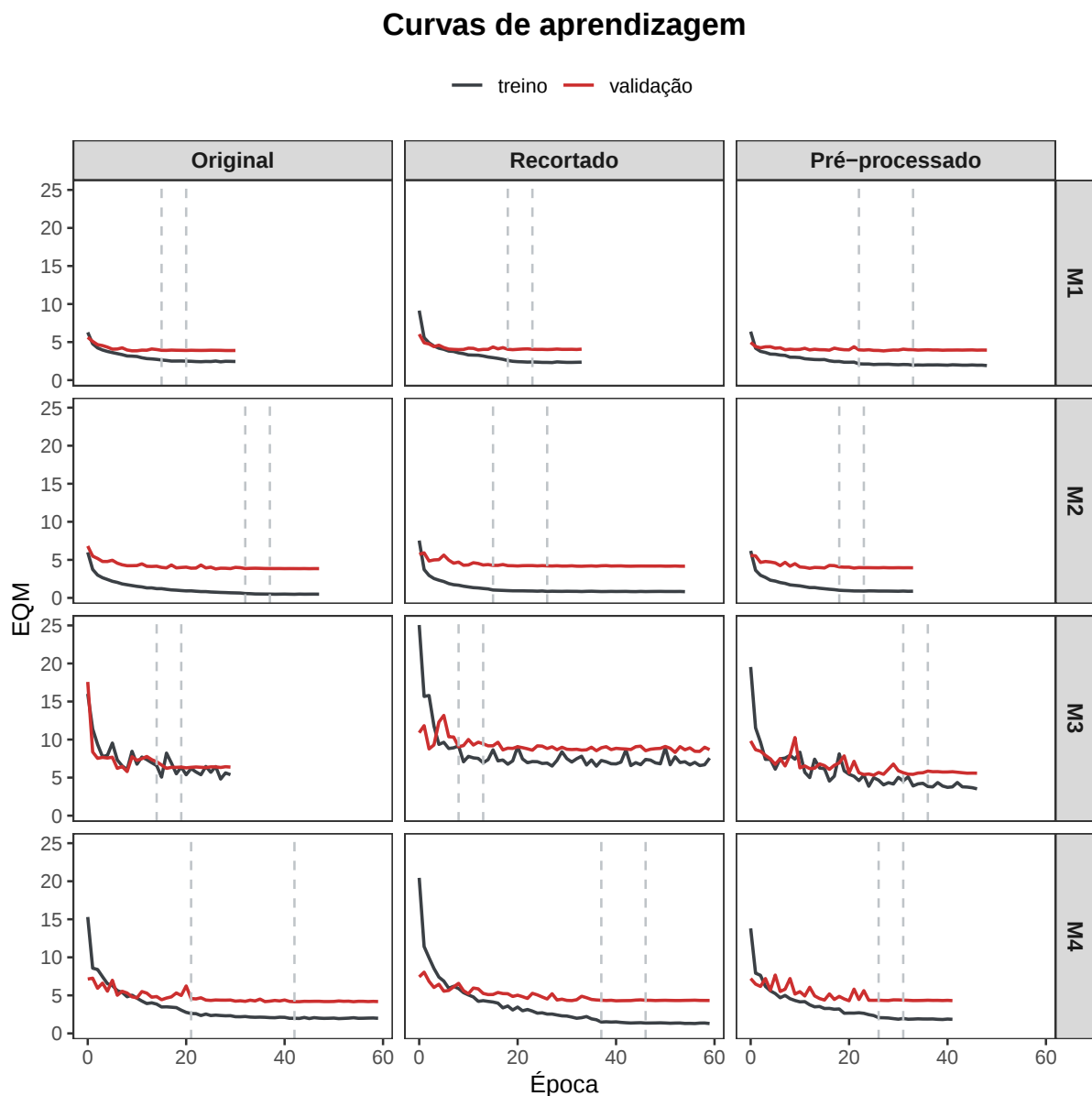


Figura 17 – Curvas de aprendizado dos modelos para os três conjuntos de imagens. As linhas verticais tracejadas indicam os pontos de alteração da taxa de aprendizado.

De modo geral, com exceção do modelo M3, que apresenta curvas de aprendizagem mais instáveis, observa-se comportamento convergente e estável para a maioria dos modelos e conjuntos de imagens, com redução do EQM nas épocas iniciais e posterior estabilização dos valores ao longo das últimas épocas.

4.3 AVALIAÇÃO DOS MODELOS

Os valores de EQM, $\overline{CV}$ e R^2 obtidos para os quatro modelos e três tipos de processamento de imagem são apresentados nas Tabelas 3 e 4. As Figuras A1 e A2 do Apêndice A ilustram os gráficos de dispersão entre os valores observados e preditos. As Figuras A3 e A4 do Apêndice A apresentam os gráficos de dispersão dos resíduos em função dos valores preditos.

Tabela 3 – Desempenho dos modelos sem *Test-Time Augmentation*.

Métrica	Modelo	Original	Recortado	Pré-processado
EQM	M1	3,2081	3,3503	2,9929
	M2	3,1190	3,1888	3,2719
	M3	4,4460	5,9386	3,8096
	M4	3,3105	3,0778	3,3672
$\overline{CV}$	M1	0,1004	0,1002	0,0963
	M2	0,0999	0,0964	0,1006
	M3	0,1316	0,1311	0,1132
	M4	0,0999	0,0988	0,1042
R^2	M1	0,8389	0,8317	0,8497
	M2	0,8433	0,8398	0,8357
	M3	0,7767	0,7017	0,8086
	M4	0,8337	0,8454	0,8309

Nota: EQM: erro quadrático médio; $\overline{CV}$: coeficiente de variação médio; R^2 : coeficiente de determinação. M1: padrão; M2: sobreamostragem; M3: ponderação; M4: híbrido.

Tabela 4 – Desempenho dos modelos com *Test-Time Augmentation*.

Métrica	Modelo	Original	Recortado	Pré-processado
EQM	M1	3,0685	3,4185	2,8017
	M2	2,9919	2,9721	2,9187
	M3	3,7674	4,5804	3,7228
	M4	2,9445	2,9453	3,0381
$\overline{CV}$	M1	0,0954	0,0971	0,0929
	M2	0,0945	0,0925	0,0941
	M3	0,1132	0,1180	0,1107
	M4	0,0944	0,0936	0,0986
R^2	M1	0,8459	0,8283	0,8593
	M2	0,8497	0,8507	0,8534
	M3	0,8108	0,7699	0,8130
	M4	0,8521	0,8521	0,8474

Nota: EQM: erro quadrático médio; $\overline{CV}$: coeficiente de variação médio; R^2 : coeficiente de determinação. M1: padrão; M2: sobreamostragem; M3: ponderação; M4: híbrido.

Sem TTA, o modelo M1 apresentou o melhor desempenho no conjunto pré-processado, com menor EQM (2,9929), menor $\overline{CV}$ (0,0963) e maior R^2 (0,8497). O modelo M3 apresentou os maiores valores de erro, especialmente no conjunto recortado. Com a aplicação de TTA, observou-se redução do EQM e do $\overline{CV}$ e aumento do R^2 na maioria dos cenários. O melhor desempenho global foi obtido pelo modelo M1 no conjunto pré-processado com TTA (EQM = 2,8017, $\overline{CV}$ = 0,0929 e R^2 = 0,8593).

DISCUSSÃO

Este trabalho teve como objetivo geral avaliar estratégias de aprimoramento do processo de predição de idade a partir de imagens de otólitos de halibute da Groenlândia (*Reinhardtius hippoglossoides*), por meio da aplicação de técnicas de pré-processamento, modificações no procedimento de treinamento e estratégias de pós-processamento, utilizando como base os dados e o modelo de RNC proposto por Moen et al. (2018). De modo geral, os resultados indicam que o pré-processamento das imagens e a técnica de TTA na avaliação do modelo podem contribuir para melhorias no desempenho preditivo.

Observou-se que o conjunto pré-processado apresentou, em diversos cenários, desempenho superior ao conjunto original e ao conjunto apenas recortado. O modelo M1 alcançou o melhor desempenho global quando aplicado ao conjunto pré-processado, tanto sem TTA quanto com TTA, evidenciando que a remoção de ruídos de fundo, a padronização do tamanho e da resolução da imagem podem favorecer a aprendizagem da rede neural. Considerando o modelo M1 com TTA, aplicado ao conjunto pré-processado, a idade foi estimada corretamente para 181 dos 715 pares de otólitos (25%). Além disso, 464 apresentaram erro absoluto de até um ano (65%). Esses resultados são comparáveis aos reportados por Moen et al. (2018), que observaram 29% de estimativas exatas e 38% com erro de um ano, totalizando 67% das amostras dentro de ± 1 ano da idade de referência. Ressalta-se que no estudo de Moen et al. (2018) o conjunto de teste correspondeu a aproximadamente 4% da base de dados, enquanto neste trabalho foi adotada a proporção de 15%. Testes preliminares indicaram que a fração de 4% gerava alta variabilidade nas métricas em função da aleatoriedade da amostragem, motivo pelo qual se optou por um conjunto de teste maior. Além disso, o modelo M1 aplicado aos dados originais, que é a configuração mais comparável à utilizada por Moen et al. (2018), apresentou desempenho inferior ao M1 com pré-processamento (EQM: 3,2081 vs. 2,9929; R^2 : 0,8389 vs. 0,8497; $\overline{CV}$: 0,1004 vs. 0,0963), reforçando o impacto positivo da etapa de pré-processamento.

Embora modelos de RNC sejam capazes de extrair características relevantes a partir de imagens brutas, estudos prévios indicam que etapas de pré-processamento podem con-

tribuir para a melhora no desempenho desses modelos. İşgüzar et al. (2024), por exemplo, incorporaram uma etapa de segmentação das imagens de otólitos antes do treinamento da RNC, isolando a estrutura de interesse e reduzindo interferências do fundo. Esses resultados sugerem que a padronização das imagens pode favorecer o processo de treinamento e melhorar o desempenho preditivo.

O modelo M3 apresentou desempenho inferior às demais estratégias. Embora a incorporação de pesos na função de perda seja uma abordagem fundamentada em *cost-sensitive learning* para mitigar o viés em dados desbalanceados, os resultados indicam que tal ponderação introduziu instabilidade na otimização da rede. Ao atribuir penalidades elevadas aos erros nas classes de idade menos frequentes, o algoritmo de *backpropagation* pode se tornar hipersensível a variações individuais ou ruídos presentes nessas amostras escassas. Esse comportamento exacerba a variância dos gradientes, resultando em um fenômeno análogo ao problema de *exploding gradient*, o que dificulta a convergência suave do modelo e favorece o aprendizado de características fictícias (*overfitting*) em detrimento da generalização.

É importante destacar que a estrutura etária é muito dinâmica e pode variar entre anos. Modelos treinados com estratégias de desbalanceamento podem apresentar maior robustez em situações nas quais o conjunto de teste apresenta distribuição distinta da observada no treinamento. Assim, abordagens que considerem o desbalanceamento, como a sobreamostragem (BUDA; MAKI; MAZUROWSKI, 2018), podem ser relevantes em aplicações práticas de longo prazo.

A estratégia de TTA é respaldada por estudos como o de Moshkov, Mathe e Kertesz-Farkas (2020), que aplicaram rotações e espelhamentos durante a etapa de teste em arquiteturas como U-Net e Mask R-CNN, observando melhora na acurácia das predições. Esses resultados indicam que a agregação de múltiplas predições obtidas a partir de diferentes perturbações da mesma imagem pode contribuir para reduzir a sensibilidade do modelo a pequenas variações na imagem.

Foram encontradas algumas dificuldades ao longo do desenvolvimento deste trabalho. As imagens dos otólitos não seguem um padrão consistente de coleta (Figura B1 do Apêndice B). Observa-se que, entre diferentes anos de coleta, há variações nas condições de captura, como enquadramento, iluminação e posicionamento, o que pode dificultar o aprendizado do modelo.

A Figura B2 do Apêndice B ilustra outra limitação observada no procedimento automatizado de remoção de fundo. Embora a ferramenta utilizada seja eficaz na segmentação e remoção do fundo, ela não atua sobre artefatos presentes no primeiro plano, como resíduos, manchas ou impurezas aderidas ao otólito. Esses elementos permanecem na imagem após o pré-processamento e podem introduzir ruído visual adicional. Assim, apesar do protocolo de pré-processamento ter contribuído para a padronização do fundo das imagens, ele não eliminou completamente fontes de variabilidade associadas à qualidade do material biológico e às

condições de captura.

Como perspectivas para o futuro deste trabalho, destacam-se as seguintes possibilidades:

- Aplicação das estratégias avaliadas neste estudo em espécies de peixes de água doce do Brasil.
- Desenvolvimento e treinamento de uma arquitetura própria de rede neural convolucional, adaptada especificamente às características morfológicas dos otólitos.
- Avaliação do impacto da inclusão de variáveis adicionais do indivíduo, como peso e comprimento, em conjunto com as imagens dos otólitos, no desempenho dos modelos de predição de idade.

REFERÊNCIAS

- BUDA, M.; MAKI, A.; MAZUROWSKI, M. A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, v. 106, p. 249–259, 2018. Disponível em: <<https://doi.org/10.1016/j.neunet.2018.07.011>>.
- CAMPANA, S.; ANNAND, M.; MCMILLAN, J. Graphical and statistical methods for determining the consistency of age determinations. *Transactions of The American Fisheries Society - TRANS AMER FISH SOC*, v. 124, p. 131–138, 01 1995. Disponível em: <[https://doi.org/10.1577/1548-8659\(1995\)124<0131:GASMFD>2.3.CO;2](https://doi.org/10.1577/1548-8659(1995)124<0131:GASMFD>2.3.CO;2)>.
- CAMPANA, S. E. Accuracy, precision and quality control in age determination, including a review of the use and abuse of age validation methods. *Journal of Fish Biology*, v. 59, n. 2, p. 197–242, 2001. Disponível em: <<https://doi.org/10.1111/j.1095-8649.2001.tb00127.x>>.
- CARBONARA, P. et al. *HANDBOOK ON FISH AGE DETERMINATION a Mediterranean experience*. [S.l.: s.n.], 2019. ISBN ISBN 978-92-5-131176-9 © FAO, 2019.
- FRIEDLANDER, A. *Elementos de Programação Não Linear*. [s.n.], 1994. Disponível em: <<https://www.ime.unicamp.br/~friedlan/livro.pdf>>.
- GATIS, D. *rembg*. 2025. Acesso em: 19 jan. 2026. Disponível em: <<https://github.com/danielgatis/rembg>>.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>.
- GRAVES, A. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012. (Studies in Computational Intelligence). Disponível em: <<https://doi.org/10.1007/978-3-642-24797-2>>.
- Greenland Institute of Natural Resources. *The greenland halibut*. s.d. Acesso em: 6 jun. 2025. Disponível em: <<https://natur.gl/arter/the-greenland-halibut/?lang=en>>.
- HAYKIN, S. *Neural networks: a comprehensive foundation*. [S.l.]: Prentice Hall PTR, 1994.
- IPARRAGUIRRE-VILLANUEVA, O. et al. Convolutional neural networks with transfer learning for pneumonia detection. *International Journal of Advanced Computer Science and Applications*, v. 13, 2022. Disponível em: <<https://doi.org/10.14569/IJACSA.2022.0130963>>.

- MA, T.-H. et al. Enhancing machine learning-based age estimation for pacific bluefin tuna: An approach with data imputation and image augmentation strategies. *Fisheries Research*, v. 274, p. 106992, 06 2024. Disponível em: <<https://doi.org/10.1016/j.fishres.2024.106992>>.
- MICHAEL, P. F.; YOON, H.-J. Survey of image denoising methods for medical image classification. In: HAHN, H. K.; MAZUROWSKI, M. A. (Ed.). *Medical Imaging 2020: Computer-Aided Diagnosis*. SPIE, 2020. v. 11314, p. 113143R. Disponível em: <<https://doi.org/10.1117/12.2549695>>.
- MOEN, E. et al. Automatic interpretation of otoliths using deep learning. *PLOS ONE*, Public Library of Science, v. 13, n. 12, p. 1–14, 12 2018. Disponível em: <<https://doi.org/10.1371/journal.pone.0204713>>.
- MOSHKOV, N.; MATHE, B.; KERTESZ-FARKAS, A. Test-time augmentation for deep learning-based cell segmentation on microscopy images. *Scientific Reports*, v. 10, 03 2020. Disponível em: <<https://doi.org/10.1038/s41598-020-61808-3>>.
- POLITIKOS, D. V. et al. Automating fish age estimation combining otolith images and deep learning: The role of multitask learning. *Fisheries Research*, v. 242, p. 106033, 2021. Disponível em: <<https://doi.org/10.1016/j.fishres.2021.106033>>.
- QIN, X. et al. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern Recognition*, Elsevier BV, v. 106, p. 107404, 2020. ISSN 0031-3203. Disponível em: <<http://doi.org/10.1016/j.patcog.2020.107404>>.
- REDMON, J. et al. You only look once: Unified, real-time object detection. *CoRR*, 2015. Disponível em: <<http://doi.org/10.48550/arXiv.1506.02640>>.
- SIGURÐARDÓTTIR, A. R. et al. Otolith age determination with a simple computer vision based few-shot learning method. *Ecological Informatics*, v. 76, p. 102046, 2023. Disponível em: <<https://doi.org/10.1016/j.ecoinf.2023.102046>>.
- SZEGEDY, C. et al. *Rethinking the Inception Architecture for Computer Vision*. 2015. Disponível em: <<https://doi.org/10.48550/arXiv.1512.00567>>.
- WANG, C.-H.; WALTHER, B. D.; GILLANDERS, B. M. Introduction to the 6th international otolith symposium. *Marine and Freshwater Research*, CSIRO Publishing, v. 70, n. 12, p. i–iii, 2019. Disponível em: <https://doi.org/10.1071/MFv70n12_ED>.
- İŞGÜZAR, S. et al. Fishagepredictionet: A multi-stage fish age prediction framework based on segmentation, deep convolution network, and gaussian process regression with otolith images. *Fisheries Research*, v. 271, p. 106916, 2024. Disponível em: <<https://doi.org/10.1016/j.fishres.2023.106916>>.

RESULTADOS COMPLEMENTARES

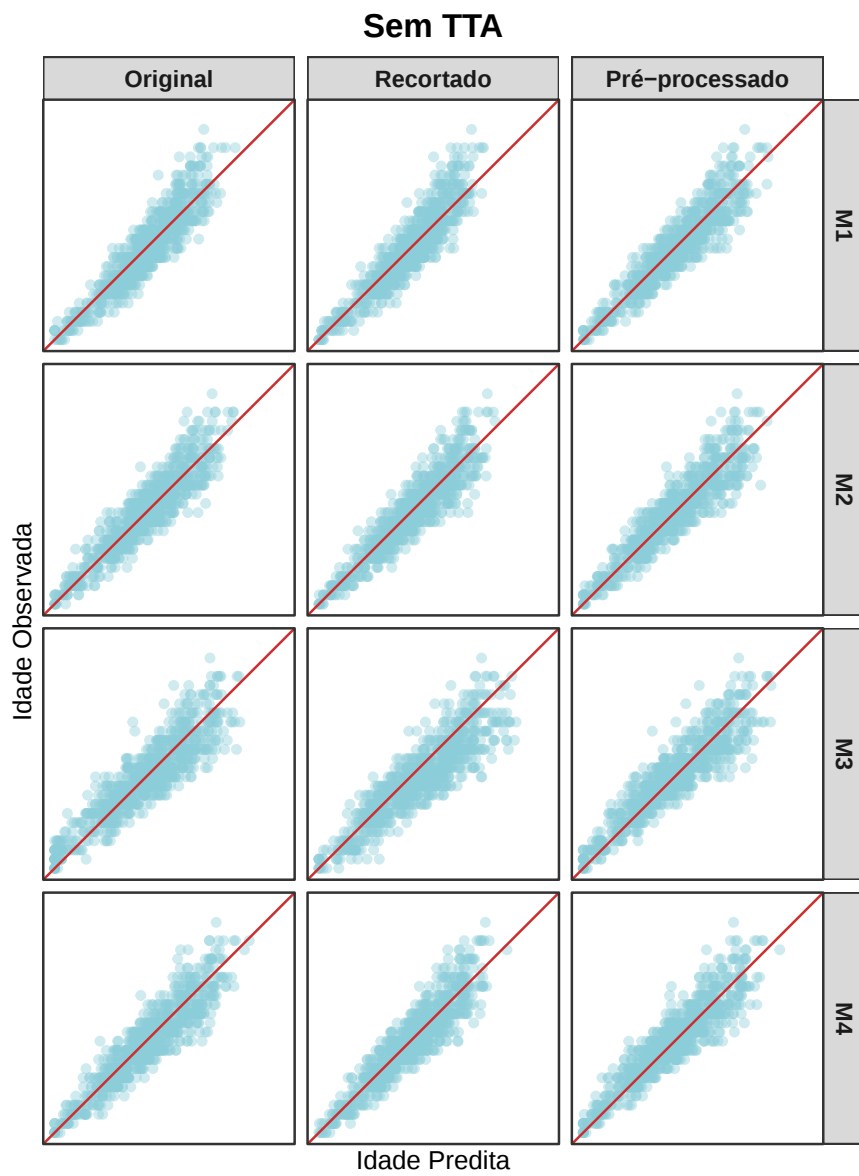


Figura A1 – Dispersão dos valores observados *versus* valores preditos sem *Test-Time Augmentation*.

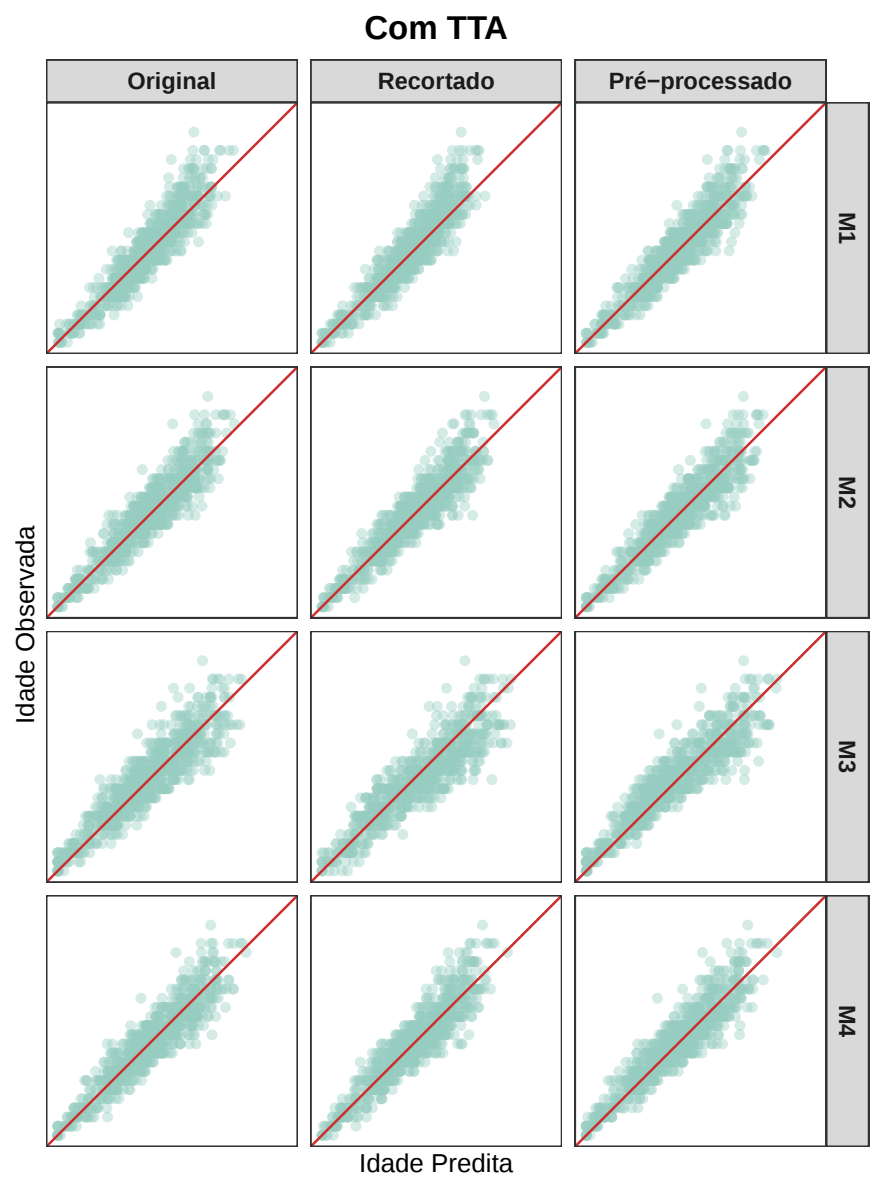


Figura A2 – Dispersão dos valores observados *versus* valores preditos com *Test-Time Augmentation*.

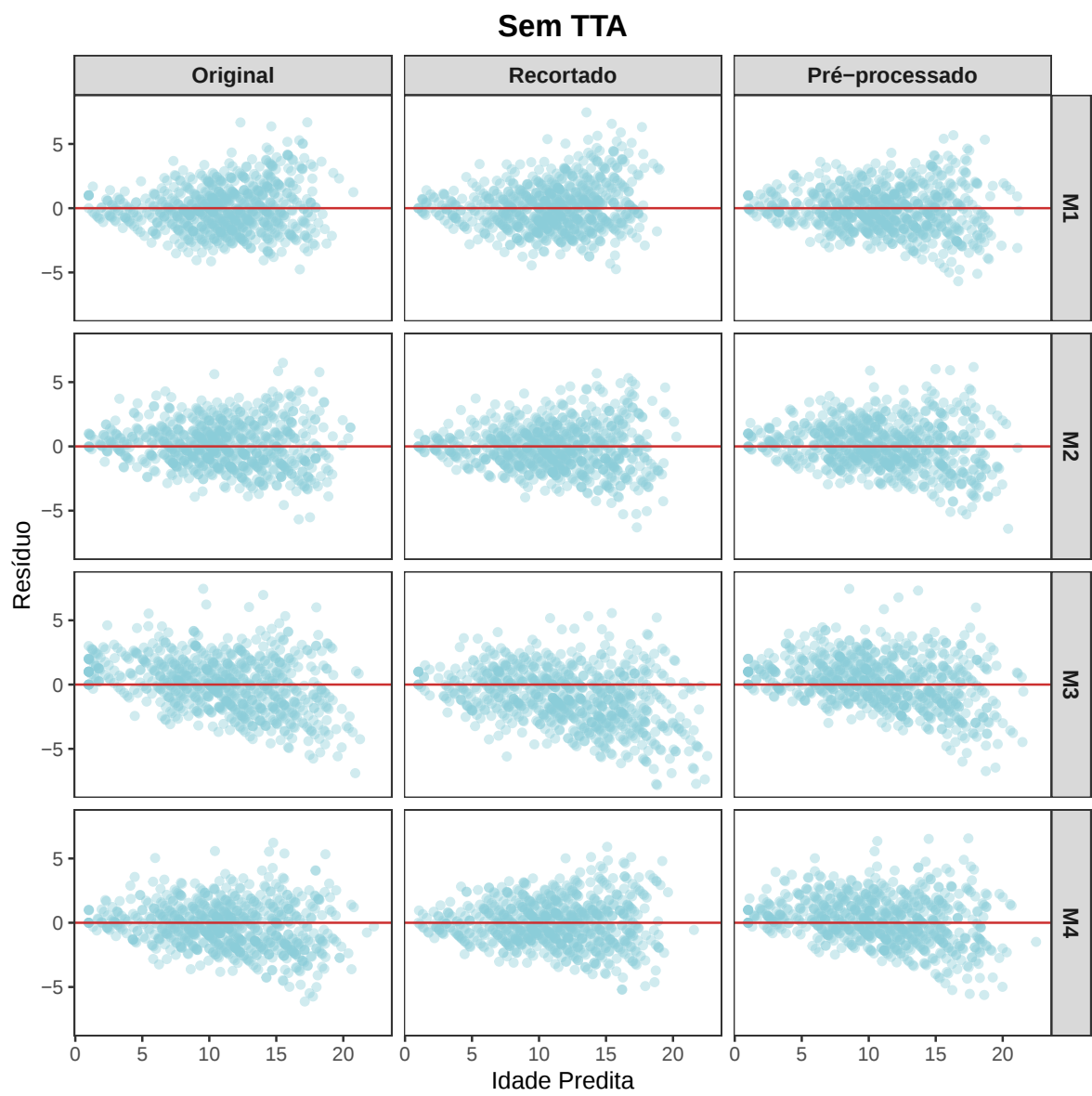


Figura A3 – Dispersão dos resíduos *versus* valores preditos sem *Test-Time Augmentation*.

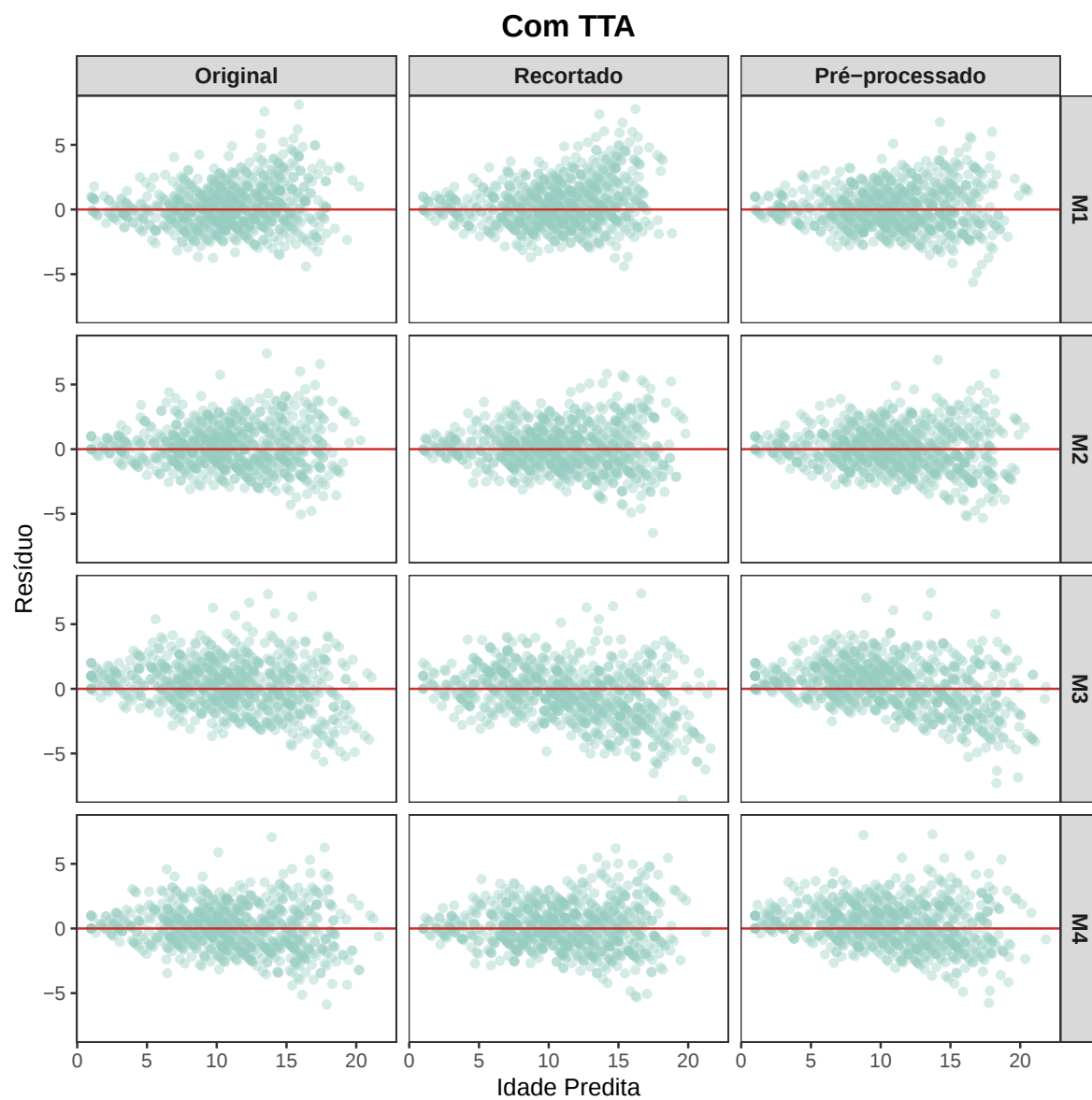
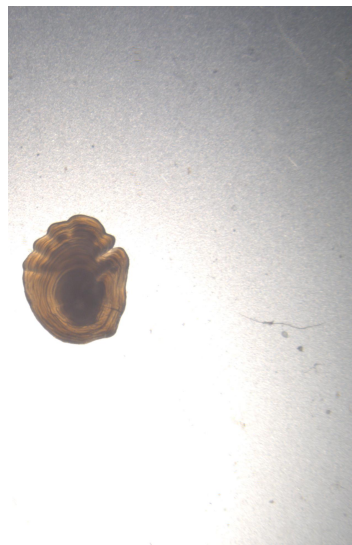
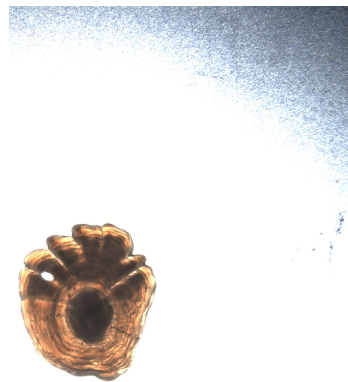


Figura A4 – Dispersão dos resíduos *versus* valores preditos com *Test-Time Augmentation*.

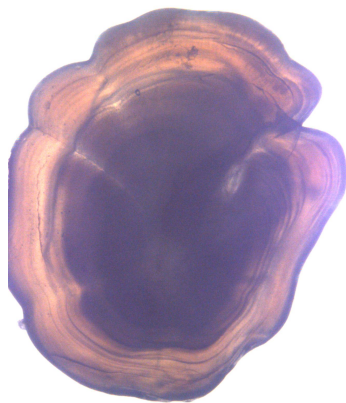
EXEMPLOS ESPECÍFICOS



(a) 2006



(b) 2008



(c) 2016



(d) 2016

Figura B1 – Imagens de otólitos coletados em diferentes anos. As imagens (a), (b) e (c) correspondem a indivíduos com 3 anos de idade, enquanto a imagem (d) corresponde a um indivíduo com 10 anos. Fonte: (MOEN et al., 2018).

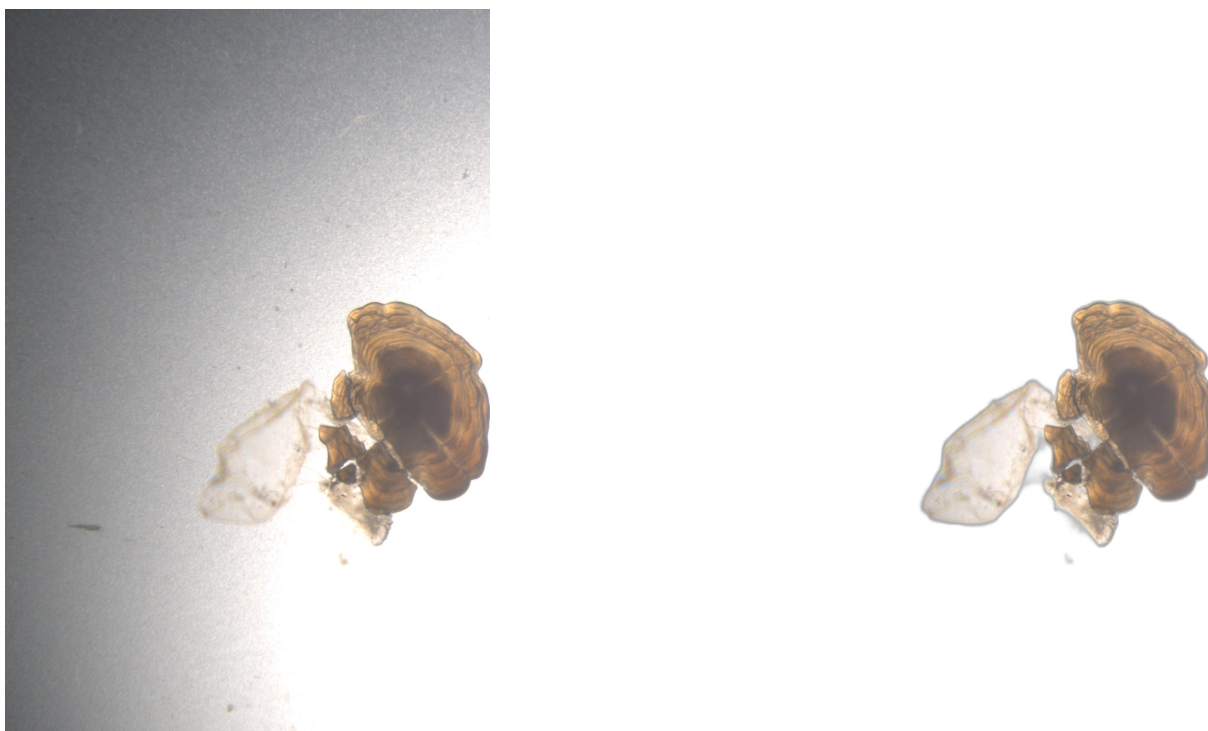


Figura B2 – Exemplo de limitação do procedimento automatizado de remoção de fundo. À esquerda, imagem original do otólito; à direita, imagem após a remoção do plano de fundo. Observa-se que artefatos presentes no primeiro plano, como resíduos e impurezas, não são removidos pelo processo.