
**Contagem e Estimativa da Massa de
Alevinos de Pintado Real Utilizando
Técnicas de Visão Computacional e
Aprendizagem de Máquina**

Adair da Silva Oliveira Junior

Contagem e Estimativa da Massa de Alevinos de Pintado Real Utilizando Técnicas de Visão Computacional e Aprendizagem de Máquina

Adair da Silva Oliveira Junior

Orientador: *Hemerson Pistori*

Dissertação apresentada ao programa de Pós Graduação, Mestrado Acadêmico em Ciência da Computação, da Universidade Federal de Mato Grosso do Sul, como requisito à obtenção do título de Mestre em Ciência da Computação.

UFMS - Campo Grande

Março/2019

Aos meus pais,
Adair e Dagumar,

A minha madrasta,
Isabel,

Aos meus irmãos,
Bruna, Lavínia e Luiz.

Agradecimentos

A todos os membros do grupo INOVISÃO, em especial à Vanessa Weber, ao Fabrício Weber, ao Gilberto Astolfi e ao meu orientador Hemerson Pistori.

A todos os meus familiares, especialmente à minha mãe, Dagumar Cardoso, e ao meu pai, Adair Oliveira.

Este trabalho recebe apoio da Universidade Católica Dom Bosco, UCDB, do Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPQ, da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, CAPES, e da Fundação de Apoio ao Desenvolvimento do Ensino, Ciência e Tecnologia do Estado de Mato Grosso do Sul, Fundect, através do termo de outorga 131/2016.

Abstract

The production of fish has undergone a significant increase in the last decades, however, it is necessary technological development to make this activity more sustainable and to leverage even more its growth. The objective of this work is to continue the development of an accountant of fingerlings, originated from the partnership between the UCDB, the IFMS and the company Projeto Pacu, which performs the automatic counting of fingerlings through the techniques of Gaussian mixture, connected components detection and Kalman filter. A fingerlings count experiment was carried out to evaluate the performance of the current counting algorithm for several parameter configurations, which, in the best combination, obtained 5.99 units of average fingerlings of the absolute errors. An experiment was also carried out to estimate the fry mass. We tested the classification and regression algorithms IBK with $k = 3$, SMO and J48 and the linear regression algorithm. In the classification approach, the J48 algorithm obtained the best result, reaching an accuracy of 58.2%. In the regression approach, the linear regression algorithm obtained the best result, reaching 1.5 grams of mean absolute errors. The experiment was extended to the convolutional neural networks for fry classification, in which the ResNet50 architecture obtained the best result, with 68.33% accuracy in the test set.

Resumo

A produção de peixes tem sofrido um significativo aumento nas últimas décadas, no entanto, é necessário desenvolvimento tecnológico para tornar esta atividade mais sustentável e alavancar ainda mais o seu crescimento. O objetivo deste trabalho é dar continuidade ao desenvolvimento de um contador de alevinos, originado da parceria entre a UCDB, o IFMS e a empresa Projeto Pacu, que realiza a contagem automática de alevinos através das técnicas de mistura de gaussianas, detecção de componentes conexos e filtro de Kalman. Foi realizado um experimento de contagem de alevinos com o intuito de avaliar a performance do atual algoritmo de contagem, para diversas configurações de parâmetros, o qual, na melhor combinação, obteve 5,99 unidades de alevinos de média dos erros absolutos. Foi realizado também um experimento de estimativa da massa de alevinos. Foram testados os algoritmos classificação e regressão IBK, com $k = 3$, SMO e J48 e o algoritmo de regressão linear. Na abordagem de classificação, com algoritmos rasos, o algoritmo J48 obteve o melhor resultado, alcançando uma acurácia de 58,2%. Já na abordagem de regressão, o algoritmo regressão linear obteve o melhor resultado, atingindo 1,5 grama de média dos erros absolutos. O experimento se estendeu no teste de redes neurais convolucionais para classificação dos alevinos, no qual a arquitetura ResNet50 obteve o melhor resultado, com 68.33% de acurácia no conjunto de teste.

Sumário

<i>Abstract</i>	i
Resumo	ii
Sumário	iii
Lista de Figuras	v
1 Introdução	1
2 Contador de Alevinos	5
3 Fundamentação Teórica	9
3.1 Visão Computacional	9
3.2 Segmentação	10
3.2.1 Limiarização	11
3.2.2 SLIC Superpixels	11
3.2.3 Subtração de Fundo	12
3.3 Filtro de Kalman	13
3.4 Aprendizagem de Máquina	14
3.4.1 Máquinas de Vetores de Suporte	16
3.4.2 C4.5	17
3.4.3 K Vizinhos Mais Próximos	19
3.4.4 Regressão Linear	19
3.5 Aprendizagem Profunda	20
3.5.1 Convolução	21
3.5.2 Redes Neurais Convolucionais	21

4 Artigo 1	25
5 Artigo 2	33
6 Considerações Finais	43
6.1 Principais Resultados	43
6.2 Trabalhos Futuros	44
Referências	46

Lista de Figuras

2.1	Protótipo atual da estrutura do contador de alevinos.	6
2.2	<i>Software</i> do contador de alevinos.	7
3.1	Exemplo de segmentação por limiarização. Da esquerda para direita, imagem original, imagem binária obtida através da limiarização e imagem com o objeto separado do fundo através de uma operação de E lógico entre a imagem original e a binária.	11
3.2	Exemplo da aplicação da técnica SLIC Superpixels, com $K = 300$, à imagem de alevinos de Pintado-Real.	13
3.3	Exemplo da aplicação da subtração de fundo.	14
3.4	Aplicação do filtro de Kalman ao contador de alevinos.	15
3.5	Exemplo de transformação de um espaço não linearmente separável para um linearmente separável.	17
3.6	Exemplo de uma árvore de decisão aplicada ao problema de classificação de alevinos por massa.	18
3.7	Exemplo da aplicação da convolução em uma imagem hipotética, canal único e dimensões 5x5, com um filtro de detecção de bordas horizontais de sobel, com dimensões de 3x3.	22
3.8	Exemplo de <i>max pooling</i> , filtro 2x2.	23
3.9	Arquitetura tradicional de um rede neural convolucional. Composta por uma camada de entrada RGB, camadas de convolução, camadas <i>pooling</i> e camadas perceptron totalmente conexas.	23

Introdução

O peixe é a segunda proteína animal mais produzida no mundo, porém, o grande crescimento populacional e a necessidade dos seres humanos de obterem uma alimentação cada vez mais saudável, fazem com que haja a necessidade de avanços na produção piscícola, principalmente no Brasil, país com produção em ascensão (FAO, 2017; Brabo et al., 2016). A exploração da pesca extrativa, atividade que possui maior produção atualmente, aliada à degradação ambiental, gerou um desequilíbrio nas populações de peixes. Este desequilíbrio trouxe, a algumas espécies, o risco de extinção, que tem resultado em grandes restrições, causando limitações a sua produção. A produção piscícola é uma atividade sustentável devido a propagação artificial de alevinos, seja ela de reprodução natural ou induzida. Há mais de 22.000 espécies de peixes catalogadas, segundo Buckup et al. (2007) no Brasil são mais de 2.000 espécies de água doce, trazendo à piscicultura uma grande quantidade de potenciais espécies para cultivo, diferenciando-a das atividades como suinocultura e avicultura onde o número de espécies para cultivo é reduzido (Andrade and Yasui, 2003).

Estima-se que o Brasil terá um crescimento de 100% em sua produção de pesca e aquicultura até o ano de 2025 (FAO, 2016b). Esse crescimento previsto será o maior da região, seguido pelo México e Argentina, com 54% e 53% de crescimento, respectivamente. Segundo dados da Embrapa, o Brasil possui 8.500

km de costa marítima e cerca de 12% do total da água doce do mundo, ou seja, possui condições favoráveis para a pesca e piscicultura. O Brasil também possui condições climáticas ideais, em grande parte do seu território, para a produção piscícola, principalmente para produção da espécie tilápia, umas das espécies de maior consumo no mundo. Outros fatores que contribuem para esse crescimento previsto, segundo a Organização das Nações Unidas para Alimentação e Agricultura (FAO), são os investimentos em pesquisa e inovação nos últimos anos (FAO, 2016a; Sidonio et al., 2012).

Devido aos fatos apontados, é evidente a necessidade de desenvolvimento tecnológico para dar suporte à produção de peixes, principalmente à piscicultura, atividade que deve ultrapassar a pesca extrativa em poucos anos. Esse suporte deve servir não só para o aumento da produção, mas também para garantir a qualidade da mesma.

Há uma dificuldade no manejo dos peixes, pois os mesmos são muito sensíveis, não podem, por exemplo, ser expostos à luz por muito tempo. Por isso, o desenvolvimento tecnológico deve ser guiado tendo como base essas limitações, além da preocupação de operação dos equipamentos sob a água ou em ambientes molhados.

A visão computacional tem como objetivo o emprego de técnicas computacionais para a extração de informações úteis a partir de imagens (Prince, 2012). Zion (2012) aponta que a visão computacional pode dar suporte em diversas atividades da piscicultura, como por exemplo, contagem, medição, estimativa da massa, identificação de sexo, avaliação da qualidade das cores de peixes ornamentais, identificação de espécies e monitoramento do bem-estar. A aplicação da visão computacional à piscicultura se torna viável, uma vez que a mesma pode reduzir o estresse dos peixes, pois, diminui a necessidade da utilização de redes e a exposição dos mesmos à luz.

A aprendizagem de máquina, ou aprendizagem automatizada, subárea da inteligência artificial, possui técnicas computacionais para o reconhecimento de padrões (Nasrabadi, 2007). Essas técnicas podem ser utilizadas em conjunto com as de visão computacional, resolvendo assim, diversos problemas de classificação e regressão, tendo como premissa, o aprendizado através de exemplos.

A regressão linear, uma das técnicas de aprendizagem de máquina, tem como objetivo encontrar modelos matemáticos lineares que correlacionem variáveis de

entrada e de saída (Mao and Monahan, 2018). Esta técnica permite a descoberta de quais características em uma imagem melhor se correlacionam com o problema em questão.

Foi criado no ano de 2001, na Universidade Católica Dom Bosco (UCDB) o grupo de pesquisa INOVISÃO, sendo composto por pesquisadores, alunos de graduação, mestrado e doutorado, o qual busca, através da inovação com técnicas de visão computacional, o desenvolvimento científico e tecnológico, visando a resolução de problemas nas mais diversas áreas, como por exemplo, o agronegócio e a perícia criminal.

O grupo INOVISÃO possui um projeto denominado FISHCV, que tem como objetivo o desenvolvimento de tecnologias voltadas à piscicultura. Hoje, este projeto está envolvido com trabalhos de classificação, contagem e estimativa da massa de alevinos e contagem de ovócitos. O projeto FISHCV, em parceria com o Instituto Federal de Mato Grosso do Sul (IFMS), desenvolve um equipamento que realiza a contagem de alevinos. Para o desenvolvimento deste equipamento, o grupo conta também com a parceria da empresa Projeto Pacu, empresa piscícola sediada em Terenos, Mato Grosso do Sul, produtora de alevinos da espécie Pintado Real, espécie híbrida desenvolvida pela própria empresa. A empresa parceira cede o ambiente, os alevinos e parte do material necessário para a coleta de imagens e a execução dos experimentos.

As informações geradas pela contagem e estimativa da massa dos alevinos auxiliarão o produtor em diversas partes do processo de criação. A contagem auxilia no ato da venda, onde os alevinos precisam ser depositados em caminhões tanques ou em caixas, trabalho que é exaustivo ao ser humano e hoje é executado manualmente. A estimativa da massa pode auxiliar o produtor em tomadas de decisão em relação à criação dos alevinos como, por exemplo, saber a necessidade do aumento na quantidade de ração administrada aos tanques, melhorando assim as taxas de crescimento dos mesmos. A geração de relatórios de venda é outro atrativo deste equipamento. Esses relatórios servem para que seus clientes tenham informações referentes aos produtos adquiridos, e assim, obtém-se uma maior credibilidade ao piscicultor.

O objetivo geral deste trabalho é dar continuidade ao desenvolvimento do equipamento de contagem de alevinos, supracitado, realizando testes de combinações de parâmetros. Foi realizado também experimentos de estimativa da massa de

alevinos, baseado em visão computacional e aprendizagem de máquina, com algoritmos de classificação e regressão IBK, SMO e J48 e o algoritmo de regressão linear. O experimento de estimativa da massa foi estendido na classificação com redes neurais convolucionais.

Este trabalho encontra-se organizado em seis capítulos. No segundo capítulo será apresentado, resumidamente, o equipamento contador de alevinos. O terceiro capítulo é composto pela fundamentação teórica, onde serão abordados os temas que se relacionam com este trabalho, sendo eles: visão computacional, aprendizagem de máquina e aprendizado profundo. Os capítulos quatro e cinco são constituídos por dois artigos, que possuem trabalhos correlatos, metodologia, resultados, discussão e resumo de dois experimentos, sendo um sobre contagem e o outro de estimativa da massa de alevinos de pintado real. No sexto, e último capítulo, serão feitas as considerações finais e propostas para trabalhos futuros.

Contador de Alevinos

A UCDB, em parceria com o IFMS e a empresa Projeto Pacu, desenvolve um equipamento que realiza a contagem automática de alevinos. O desenvolvimento deste se iniciou devido a necessidade da empresa por um equipamento que realizasse a contagem dos alevinos, na translocação dos mesmos dentro da empresa, entre os tanques de criação e no momento da venda, visto que as contagens eram feitas de maneira manual.

Nas transferências entre tanques, atualmente, é realizada uma estimativa utilizando uma peneira. É feita uma multiplicação entre a quantidade de alevinos que preenchem uma peneira e a quantidade de peneiras totalmente preenchidas que são retiradas de um tanque. No momento da venda, aproximadamente 9 funcionários realizam a contagem, onde cada um fica responsável por preencher uma caixa, geralmente com 100 unidades de alevinos cada. Os dois processos são exaustivos e não trazem precisão à criação de alevinos, motivos que justificam o desenvolvimento do equipamento proposto.

O equipamento consiste de 4 componentes principais sendo eles: a estrutura, a iluminação, a câmera e o *software*. A estrutura é montada de maneira que fique inclinada em torno de 12 graus, aproximadamente, por onde os alevinos deslizam com o auxílio de água. A inclinação e a vazão da água, influenciam na velocidade em que os alevinos deslizam sobre a estrutura. Se a velocidade for

alta, as imagens captadas podem não ser suficientes para realizar o rastreamento dos alevinos, por outro lado, se for baixa, o equipamento terá uma pequena taxa, alevinos por hora, de contagem.

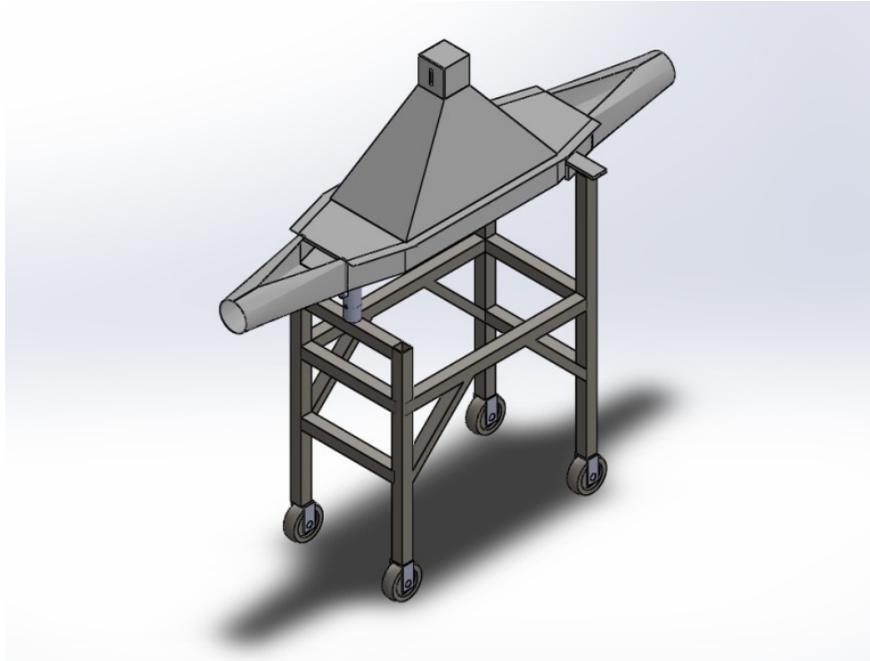


Figura 2.1: Protótipo atual da estrutura do contador de alevinos.

A iluminação é também um fator importante e deve causar pouca reflexão, bem como corrigir alterações exteriores, fazendo com que as imagens sejam uniformes, otimizando o foco e o contraste. A câmera é acoplada à estrutura e realiza a captura das imagens. Os parâmetros de taxa de quadros captados por segundo e a resolução também influenciam diretamente o sistema de contagem, uma vez que se a taxa for pequena, não haverá quadros suficientes para o rastreamento dos alevinos, em contrapartida, se a taxa for alta, acarretará em um maior consumo dos recursos do computador, podendo até impossibilitar a contagem em tempo real. O *software* opera em um computador, atualmente em um *laptop* por razões de mobilidade, responsável pelo processamento das imagens, e portanto, da contagem.

Ao longo do desenvolvimento do equipamento, tanto a estrutura quanto o *software* passaram por melhorias. Na versão atual, a iluminação é instalada na parte de baixo da estrutura, sendo difundida por uma placa leitosa. A nova estrutura possui também uma tampa que minimiza a influência da iluminação externa, trazendo assim maior uniformidade às imagens. O equipamento conta com uma

câmera com resolução *full HD* (1920 x 1080 *pixels*) e uma taxa de captura de 30 quadros por segundo. A Figura 2.1 e a Figura 2.2 apresentam, respectivamente, a estrutura e a interface do *software* na atual versão do contador de alevinos.

Na Figura 2.2, podemos observar uma linha horizontal na imagem que está sendo processada. Esta linha demarca a região de contagem, ou seja, o alevino deve atravessar a mesma para que o mesmo seja contado. À esquerda, na mesma figura, é apresentada a janela para alteração dos parâmetros de contagem. Ao lado da imagem que está sendo processada podemos observar o valor atual da contagem e a taxa de captura de quadros. Há também um modo simplificado, que realiza a contagem sem a apresentação dos quadros, otimizando assim o processamento.

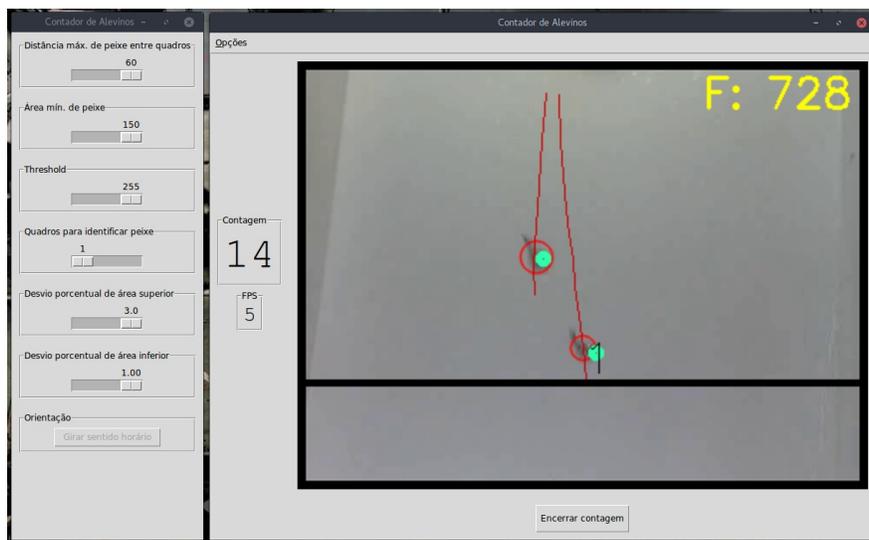


Figura 2.2: *Software* do contador de alevinos.

O *software* conta com as técnicas de segmentação por subtração de fundo (*Background Subtraction*), detecção de componentes conexas (*Blob Detection*), detecção de contornos (*Find Countours*) e filtro preditivo de *Kalman* (*Kalman Filter*). O algoritmo de contagem funciona basicamente na segmentação por subtração de fundo e na detecção de regiões de interesse e contornos, que são rastreados através das predições realizadas pelo filtro de *Kalman*. O mesmo possui parâmetros que definem o padrão da contagem, como por exemplo a área mínima e o valor de limiarização para detecção das regiões de interesse, no caso os alevinos, e desvio padrão da área dos mesmos, para correção quando há aglomerações. Em linhas gerais, o algoritmo realiza a contagem através dos seguintes passos:

1. Segmentação por subtração de fundo;
 2. Detecção de regiões de interesse e contornos, para rotulação dos mesmos como candidatos a alevinos;
 3. Aplicação do filtro de *Kalman* aos candidatos e associação às regiões de interesse e contornos detectados no novo quadro.
 4. Verificação da quantidade de vezes que um candidato foi rastreado. Se alcançar o valor de um parâmetro, denominado limiar candidato, é feita a mudança da rotulação de candidato para alevino.
 5. Verificação da posição do alevino em relação à região de contagem. Se o mesmo a ultrapassou é realizada o cômputo.
- Distância entre *blobs*: distância mínima entre os *blobs* de um quadro. Se um *blob* for encontrado dentro desta distância, os dois são associados como um único *blob*.
 - Área do *blob*: área mínima de um *blob*. *Blobs* com áreas menores são descartados.
 - Limiar *blob*: valor para segmentação por limiarização do *blob*.
 - Limiar candidato: quantidade de vezes que um candidato deve ser rastreado para se tornar um alevino.
 - Desvio padrão: desvio padrão. Em casos de aglomeração é feita uma análise da área do *blob* para definição se há mais de um alevino.
 - Área mínima: área mínima necessária para um *blob* ser contado como um alevino. Semelhante à área do *blob*, porém *blobs* menores podem ser detectados para questões de rastreamento, sendo feita uma verificação na hora da contagem.

O *software* necessita de melhorias quanto à calibração dos parâmetros, que hoje é feita manualmente pelo usuário, o qual necessita de experiência com os mesmos. A ideia é facilitar este processo para que qualquer usuário, mesmo que sem experiência, possa realizar este procedimento, se possível, automaticamente.

Fundamentação Teórica

Neste capítulo serão apresentados os conceitos necessários para a realização deste trabalho.

3.1 Visão Computacional

Segundo Prince (2012), a visão computacional tem como objetivo o emprego de técnicas computacionais para a extração de informações úteis a partir de imagens. Para Shapiro and Stockman (2001), o objetivo da visão computacional é criar decisões úteis sobre objetos reais e cenas através do uso de imagens.

A visão computacional se apresentou uma tarefa surpreendentemente desafiadora e tem despertado um grande interesse em milhares de pesquisadores nos últimos 40 anos. Embora grandes avanços venha acontecendo, estamos longe ainda de obter um desempenho próximo ao humano, pois isto se deve a complexidade dos dados visuais.

A visão computacional possui relações com diversas outras áreas do conhecimento, sendo elas: processamento de sinais, robótica, inteligência artificial, aprendizagem de máquina, matemática, neurobiologia, ótica, entre outras. Muitas das técnicas desenvolvidas nessas áreas podem ser aproveitadas na visão computacional e podemos dizer o mesmo do oposto (Prince, 2012).

Um sistema de visão computacional consiste, geralmente, em quatro componentes básicos, sendo eles: iluminação, câmera, *hardware* e *software*. A iluminação é um componente importante, pois se a mesma for bem administrada pode reduzir reflexão, sombras e ruídos, facilitando assim as fases de pré-processamento, em alguns casos até a eliminando. A câmera tem como objetivo a captação de imagens codificadas em formato digital. Os parâmetros da câmera, como por exemplo resolução, afetam diretamente a fase de processamento, pois quanto maior a resolução, mais informações terão que ser processadas e assim, aumenta-se o tempo para conclusão desta fase. Um contra ponto é que se a resolução da câmera for baixa, as informações disponíveis para processamento podem ser insuficientes. Por último, o *hardware* e o *software* são os elementos que formam o sistema computacional que irá processar as informações das imagens captadas (Szeliski, 2010).

Como o *software* proposto também tem como objetivo estimar a massa de alevinos, é de extrema importância a preocupação com os componentes apontados. As imagens que servirão de entrada para o *software* serão capturadas em um ambiente que contém água, e a mesma poderá então causar uma série de ruídos, como por exemplo reflexão da luz. Portanto, a administração da iluminação deverá ser realizada de maneira a reduzir esses ruídos.

Podemos dividir um sistema de visão computacional em 5 principais etapas, sendo elas: aquisição de imagens, pré-processamento, extração de características, segmentação e processamento. Atualmente, a terceira etapa, extração de características, pode ser substituída por algoritmos de aprendizagem de máquina, que através de um conjunto de treinamento aprendem características automaticamente. Em alguns áreas, como a robótica, podemos ainda adicionar a etapa de atuação.

3.2 Segmentação

A segmentação é uma técnica de visão computacional que tem como objetivo associar um rótulo para cada um dos *pixels* em uma imagem, de modo que os *pixels* que pertencem ao mesmo objeto são atribuídos ao mesmo rótulo (Prince, 2012).

3.2.1 Limiarização

A limiarização, técnica de segmentação considerada simples, consiste em rotular os *pixels* de uma imagem verificando se os valores de seu espaço de cor, por exemplo, o espaço de cores RGB, respeitam os limiares impostos (Sahoo et al., 1988). Essa técnica é muito útil quando os objetos em uma imagem possuem cores que se distanciam, no espaço de cores, do fundo da imagem. A Figura 3.1 representa um exemplo de limiarização aplicado a imagem de um alevino.

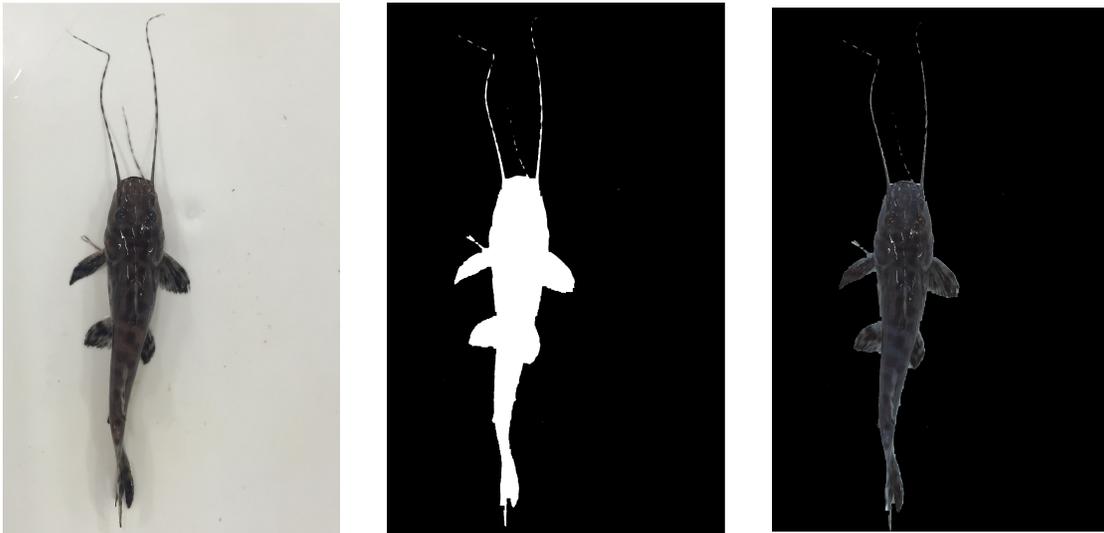


Figura 3.1: Exemplo de segmentação por limiarização. Da esquerda para direita, imagem original, imagem binária obtida através da limiarização e imagem com o objeto separado do fundo através de uma operação de E lógico entre a imagem original e a binária.

3.2.2 SLIC Superpixels

Achanta et al. (2012) apresentam em seu trabalho técnicas de *superpixels*. Essas são técnicas de segmentação que tem como objetivo agrupar *pixels* que possuem características parecidas, criando assim regiões atômicas que representam uma informação conjunta. Com a clusterização, informações redundantes são descartadas, diminuindo assim o custo computacional nas tarefas de processamento.

SLIC Superpixels é uma técnica de segmentação proposta por Achanta et al. (2012) e que possui melhor desempenho, em tempo de processamento, do que as

outras técnicas de *superpixels* existentes. Esta técnica é baseada no algoritmo de clusterização *k-means*, introduzida por Hartigan and Wong (1979).

Na técnica *k-means*, é dada uma coleção de pontos, os quais serão divididos em *k clusters*. Esta divisão é feita através do cálculo da distância Euclidiana de cada ponto até o ponto central de cada *cluster*, onde o ponto é designado para o *cluster* cuja distância calculada for a menor. Quando aplicada em uma imagem, os *clusters* tendem a ter uma forma mais rígida, se aproximando de figuras geométricas quadradas (Hartigan and Wong, 1979).

Na técnica SLIC Superpixels, além da análise da distância Euclidiana das coordenadas x e y entre os *pixels* e os centros dos *superpixels*, é feita também a análise da distância Euclidiana no espaço de cores CIELab. Portanto utiliza-se um espaço de 5 dimensões, sendo ele o espaço $[labxy]$. Nesta técnica a quantidade de *superpixels* é definida pelo parâmetro K , portanto, em uma imagem com N *pixels* cada *superpixel* tem em média N/K *pixels*. No caso de *superpixels* do mesmo tamanho, a distância entre os centros dos *superpixels* seria $S = \sqrt{N/K}$. Para isso o algoritmo define K centros de *superpixels* $C_k = [l_k, a_k, b_k, x_k, y_k]$, com $k = [1, K]$ em cada grade no intervalo S . Para cada *pixel* $P_i = [l_i, a_i, b_i, x_i, y_i]$, com $i = [1, N]$, da imagem é calculada a distância aos centros C_k através da equação 3.1:

$$\begin{aligned} d_{\text{lab}} &= \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \\ d_{xy} &= \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \\ D_s &= d_{\text{lab}} + \frac{m}{S} d_{xy} \end{aligned} \tag{3.1}$$

onde D_s representa a soma da distância no espaço CIELab, representado por d_{lab} , e a distância xy , representado por d_{xy} normalizada no plano pelo intervalo S . A variável $m \in [1, 20]$, representa a compacidade do *superpixel*, sendo 10 seu valor padrão (Achanta et al., 2012).

A Figura 3.2 representa um exemplo da técnica SLIC, com $K = 300$, aplicada à imagem de alevinos de Pintado-Real.

3.2.3 Subtração de Fundo

Outra categoria de segmentação é a subtração de fundo. Este tipo de técnica tem como objetivo gerar uma máscara de primeiro plano, binária, que pode de-

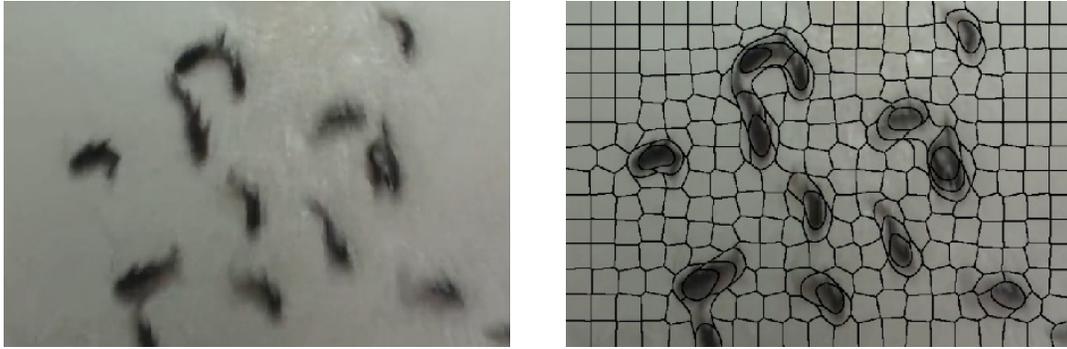


Figura 3.2: Exemplo da aplicação da técnica SLIC Superpixels, com $K = 300$, à imagem de alevinos de Pintado-Real.

tectar os *pixels* que se movem em uma cena. Esta categoria é muito utilizada na segmentação de problemas que precisam rastrear objetos em movimento a partir de câmeras estáticas, ou seja, câmeras que capturam imagens fixadas em uma única posição (Piccardi, 2004). A Figura 3.3 apresenta um exemplo da subtração de fundo. A Figura 3.3a contém o fundo estático, na Figura 3.3b há alevinos se movimentando e na Figura 3.3c é apresentada a imagem binária resultante da subtração de fundo.

3.3 Filtro de Kalman

Uma categoria de problemas muito recorrente em visão computacional é a de rastreamento de objetos em vídeos. Um exemplo de problema é o de rastreamento durante a oclusão de objetos, ou seja, problemas em que um objeto é sobreposto por um obstáculo ou outro objeto e não pode ser observado. Para resolução desses problemas, a visão computacional conta com uma coleção de algoritmos conhecidos como filtros preditivos.

O filtro de Kalman é uma técnica amplamente utilizada para estimar estados de sistemas que possuem medidas indiretas ou incertas, e pode ser utilizado como um filtro preditivo para resolução de problemas em visão computacional (Ristic et al., 2003).

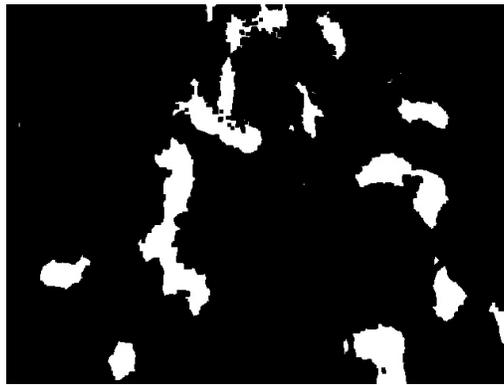
Para aplicação do filtro de Kalman, é preciso realizar o consumo de quadros de um vídeo. Após a segmentação de um quadro, é feita a localização de objetos de interesse. É aplicado então o filtro para prever em qual posição o objeto atual será encontrado no quadro subsequente. No consumo do novo quadro



(a) Fundo estático.



(b) Alevinos em movimento.



(c) Imagem binária, resultado da subtração de fundo.

Figura 3.3: Exemplo da aplicação da subtração de fundo.

é feita novamente a segmentação, e logo após, é feita associação dos objetos encontrados com a predição da posição dos objetos do quadro anterior. É feita a associação do objeto mais próximo à predição, e como o filtro pode não acertar o ponto exato onde o objeto estará, é feita a correção do mesmo com a posição atual do objeto encontrado. A Figura 3.4 apresenta o rastreamento de alevinos através da utilização do filtro de Kalman, onde as linhas vermelhas representam o histórico de predições.

3.4 *Aprendizagem de Máquina*

A aprendizagem é um fenômeno que ocorre quando um agente melhora seu desempenho em atividades que serão realizadas no futuro através de experiências passadas e observações sobre o mundo. Um tipo de aprendizagem, a aprendi-



Figura 3.4: Aplicação do filtro de Kalman ao contador de alevinos.

zagem de máquina ou aprendizagem automatizada, é realizada a partir de uma coleção e pares de dados, entrada e saída, os quais são analisados e como consequência são derivadas funções que podem prever saídas para novas entradas. As informações de entrada são formadas por um único ou um conjunto de atributos, representados de maneira fatorada, que se relacionam com saídas que podem ser valores numéricos ou valores discretos (Thrun and Norvig, 2011).

A ferramenta WEKA, implementada na linguagem de programação Java, contém um conjunto de algoritmos capazes de realizar, a partir de um conjunto de dados, a criação de funções que possam prever futuros valores. Os dados são importados pela ferramenta através de arquivos ARFF, do inglês (*Attribute-Relation File Format*), arquivos de textos que descrevem uma lista de instâncias que compartilham um conjunto de atributos. Além da criação dessas funções, a ferramenta ainda conta com funcionalidades de avaliação do modelo criado através de coeficientes de correlação, precisão, revocação, medida-f, matriz de confusão entre outros (Hall et al., 2009).

Há duas abordagens em aprendizagem de máquina, considerando-se o tipo da saída esperada, sendo elas: regressão e classificação. A regressão tem como objetivo a obtenção de um modelo matemático, que, dado valores de entrada, seja possível obter um valor de saída contínuo, de menos infinito até mais infinito.

Já a classificação, deseja obter um modelo, que, dado valores de entrada, seja possível obter um valor de saída discreto, comumente chamado de classe, que definem os dados da entrada (Hall et al., 2009).

Quanto ao aprendizado, há 3 tipos, sendo eles: supervisionado, não-supervisionado e semi-supervisionado. O aprendizado supervisionado ocorre quando um conhecimento, modelo, é aprendido a partir de exemplos, ou seja, pares de entradas e saídas. Em uma aprendizagem não-supervisionada não há exemplos, você possui apenas as entradas. Este tipo de aprendizagem é utilizada, principalmente, em problemas de agrupamento. Já o aprendizado semi-supervisionado é a junção dos dois tipos anteriores. Uma das técnicas de aprendizado semi-supervisionado é aplicada através dos seguintes passos:

- Aprender um modelo a partir de exemplos;
- Aplicar o modelo aprendido à dados não rotulados, ou seja, dados os quais não sabemos a saída;
- Utilizar esses dados, agora rotulados, para reforçar o modelo aprendido.

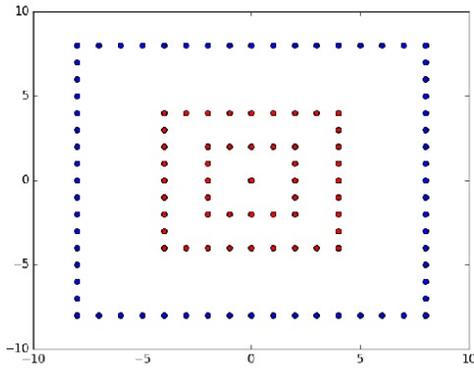
Esta técnica pode ser utilizada quando não é possível obter muitos dados rotulados ou quando a rotulação é inviável devido a grande quantidade dos mesmos (Hall et al., 2009).

3.4.1 Máquinas de Vetores de Suporte

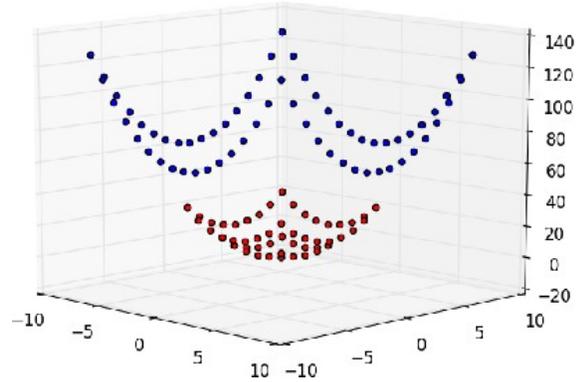
Máquinas de Vetores de Suporte (*Support Vector Machine*), conhecida como SVM, é uma técnica de classificação introduzida por Boser et al. (1992) e Vapnik and Vapnik (1998), que dado um conjunto de classes, cria-se uma função que as separe. Imaginemos um plano com dois conjuntos, positivos e negativos, uma máquinas de vetores de suporte separaria esses conjuntos com uma margem. Uma vez que podem existir diversas margens que dividem o plano, e que em cada divisão haja apenas exemplo positivos e exemplos negativos, a margem escolhida será aquela que maximize a distância da margem aos exemplos, negativo e positivo, mais próximos (Guyon et al., 2002).

Em casos em que as classes não podem ser separadas por um plano linear, é necessária a aplicação de algumas técnicas para que a SVM tenha um bom resultado. Uma dessas técnicas é a aplicação de funções *kernel*, que incluem,

por exemplo, não-linearidades gaussianas e polinomiais. A Figura 3.5 apresenta um exemplo da aplicação de uma função de transformação, onde foi adicionado um novo eixo através da Equação 3.2, tornando um conjunto de dados, que originalmente não era, linearmente separável.



(a) Não linearmente separável.



(b) Linearmente separável.

Figura 3.5: Exemplo de transformação de um espaço não linearmente separável para um linearmente separável.

$$z = y^2 + x^2 \quad (3.2)$$

Apesar de originalmente ter sido criada para problemas binários, a técnica SVM pode ser utilizada para problemas de múltiplas classes. Isso pode ser realizado através de dois modos: realizando vários testes entre duas classes ou através de testes de uma classe contra todas as outras (Ahmed et al., 2012).

Neste trabalho utilizamos o algoritmo SMO (*Sequential Minimal Optimization*), disponível no *software* WEKA, que consiste na re-implementação do algoritmo SVM descrito em Platt (1998). Esta implementação, proposta em 1998, quebra o problema de programação quadrática original em uma série de problemas menores, resultando em uma queda no tempo de computação do treinamento, tornando possível o treinamento de grandes conjuntos de testes de forma eficiente.

3.4.2 C4.5

Uma árvore de decisão é uma estrutura que consiste em nós de decisão e folhas que indicam classes. Uma instância, que deve ser classificada, percorre

a árvore da raiz, que também é um nó de decisão, até encontrar uma folha, podendo assim percorrer diversos nós. Os nós de decisão especificam testes que serão realizados, gerando ramificações para cada possível resultado. As folhas indicam a qual classe a instância, que percorreu a árvore, pertence. A Figura 3.6 apresenta um exemplo de uma árvore de decisão, aplicada ao problema de classificação dos alevinos por massa. Neste problema, temos os atributos área e perímetro do peixe e as classes abaixo do peso, peso normal e acima do peso. As formas retangulares representam os nós de decisão e as elipsoidais representam as folhas.

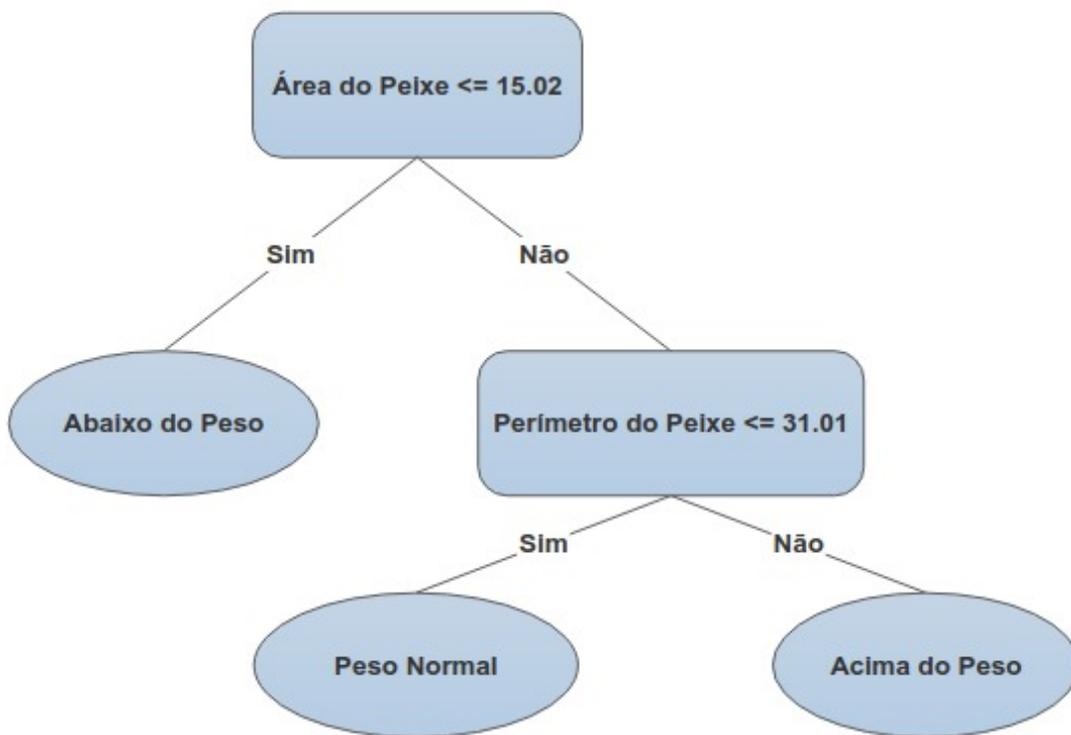


Figura 3.6: Exemplo de uma árvore de decisão aplicada ao problema de classificação de alevinos por massa.

O algoritmo C4.5 é um algoritmo que utiliza árvore de decisão para classificar um conjunto de dados. A geração da árvore de decisão é realizada através de um conjunto de casos de treinamento, utilizando a estratégia de divisão e conquista (Quinlan, 2014). Em um caso simples, onde o conjunto de treinamento pertença a uma única classe, a árvore é constituída de apenas uma folha, que identifica

esta classe. Caso contrário o conjunto de treinamento é dividido em subconjuntos, onde em cada subconjunto, todos os casos de treinamento pertençam a mesma classe. Essa divisão é feita através dos nós de decisão, os quais contêm um teste de um determinado atributo. A saída desses nós possuem ramificações que representam os subconjuntos criados a partir dos possíveis resultados para o atributo testado.

Neste trabalho, utilizamos o algoritmo J48 disponível no *software* WEKA, que consiste na re-implementação do algoritmo C4.5, descrito em Michie et al. (1993).

3.4.3 *K Vizinhos Mais Próximos*

O algoritmo K vizinhos mais próximos (*K-Nearest Neighbors*), conhecido como KNN, introduzido por Cover and Hart (1967), está entre os mais simples algoritmos de classificação. O KNN é um tipo de aprendizagem baseada em instância, também conhecido como aprendizado preguiçoso. Para realizar a classificação, é sempre necessária a utilização do conjunto de treinamento, uma vez que não há uma etapa de aprendizagem separada da de classificação.

O algoritmo KNN consiste na classificação a partir da maior frequência entre os k vizinhos mais próximos. Por este motivo, em problemas binários, é aconselhável a utilização de valores ímpares para k , uma vez que para valores pares, é possível haver um empate entre as classes e portanto, gerando uma indecisão. Em seu modo mais simples, quando $k = 1$, uma instância é classificada com o rótulo do elemento, pertencente ao conjunto de treinamento, mais próximo. Para $k > 1$ é necessário realizar a busca dos k elementos mais próximos, portanto, o valor de k influencia diretamente no custo computacional. Este algoritmo pode ser utilizado também para regressão, para tal, é calculada a média ponderada das distâncias dos vizinhos mais próximos.

Neste trabalho, utilizamos o algoritmo IBK, disponível no *software* WEKA, que consiste na re-implementação do algoritmo KNN descrito em Aha et al. (1991).

3.4.4 *Regressão Linear*

A regressão linear é uma técnica de aprendizagem com origem na estatística. Em sua forma mais básica, com apenas uma variável, dada duas variáveis em um plano, X e Y, e um conjunto de pontos que associam valores entre essas variáveis, é realizada a dedução de uma equação em função de X que corresponda

Y, portanto, a partir de qualquer valor de X é possível estimar o valor de Y. A dedução desta equação é realizada de modo a minimizar o somatório dos quadrados dos erros. A função gerada na regressão linear é dada pela equação reduzida da reta descrita na Equação 3.3, onde a é a declividade da reta e b é o valor em que a reta cruza o eixo Y (Neter et al., 1996).

$$Y = a * X + b \quad (3.3)$$

O coeficiente de Pearson, ou também conhecido como coeficiente de correlação, mede o grau de correlação entre duas variáveis. O mesmo varia de -1 à 1, sendo que quanto mais próximo do valor 1 maior a correlação direta e quanto mais próximo do valor -1 maior é a correlação inversa.

A regressão linear pode ser realizada de maneira multivariada, ou seja, não apenas entre duas variáveis, e sim n variáveis, sendo representadas em n dimensões. Diferente da regressão linear simples, esta terá como resultado não uma reta, mas sim um hiperplano.

3.5 *Aprendizagem Profunda*

A aprendizagem estruturada profunda (*Deep structured learning*), comumente chamada de aprendizagem profunda (*Deep learning*), é uma subárea da aprendizagem de máquina, que surgiu com a intenção de aproximá-la de um dos seus objetivos iniciais, a inteligência artificial (Deng et al., 2014; Lab, 2015). Na última década, as técnicas desenvolvidas, através das pesquisas sobre aprendizagem profunda, ocasionaram avanços notáveis em trabalhos de processamento de sinais e de informações, de novos escopos, incluindo aspectos chaves da aprendizagem mecânica e da inteligência artificial.

Deng et al. (2014) definem a aprendizagem profunda como uma classe de técnicas de aprendizado de máquina, que exploram muitas camadas de processamento de informações não-lineares, para extração e transformação de recursos supervisionados ou não supervisionados e para análise e classificação de padrões.

Inicialmente, criadas para processar sinais, como por exemplo, voz, a aplicação das técnicas de aprendizagem profunda em visão computacional tem alcançado um notável progresso nos últimos anos, principalmente no campo de

reconhecimento de objetos. Segundo (Russakovsky et al., 2015), atualmente, a aprendizagem profunda detêm uma taxa de erro menor que 5% no ImageNet, desafio anual de reconhecimento de objetos, alcançando assim, um resultado melhor do que ao de um ser humano para a mesma tarefa. Anteriormente a elas, a visão computacional ficou muito tempo dependente de técnicas como SIFT e HOG, utilizadas para capturar vetores de atributos para posteriormente serem utilizados em algoritmos de classificação, como SVM, C4.5 e KNN (Deng et al., 2014).

3.5.1 Convolução

Em processamento de imagens, a convolução é o processo de soma das multiplicações entre cada *pixel* da imagem com seus vizinhos locais, definidas por um núcleo (*kernel*), também conhecido como matriz de convolução. Esta técnica possui diversas aplicações, sendo as principais: detecção de bordas, suavização e extração de atributos (Prince, 2012; Shapiro and Stockman, 2001; Parker, 2010). A Figura 3.7 apresenta a aplicação da convolução em uma imagem hipotética, com canal único e dimensões 5x5, com um filtro de detecção de bordas horizontais de Sobel, de 3x3.

Neste exemplo, a abordagem utilizada, após a aplicação do filtro, ocasionou na redução das margens da imagem original, fazendo que a imagem de 5x5 se transformasse em uma de 3x3. Porém há outras abordagens para aplicação de uma convolução. Uma delas é o espelhamento dos *pixels* próximos às bordas, proporcionalmente ao tamanho do filtro, para que após a aplicação do mesmo a imagem mantenha seu tamanho original.

3.5.2 Redes Neurais Convolucionais

As redes neurais convolucionais (*Convolutional Neural Networks*), CNNs, são redes inspiradas biologicamente, variantes das redes multicamadas Perceptron (*Multilayer Perceptron*), MLPs (LeCun et al., 1990). Essas redes foram projetadas para processar dados de duas ou mais dimensões, como por exemplo, imagens e vídeos (Arel et al., 2010). Baseadas na estrutura do sistema visual humano, foram introduzidas por Fukushima and Miyake (1982), sendo os primeiros a utilizarem modelos computacionais baseados na conectividade local entre os neurônios e em transformações da imagem hierarquicamente. Posteriormente, LeCun

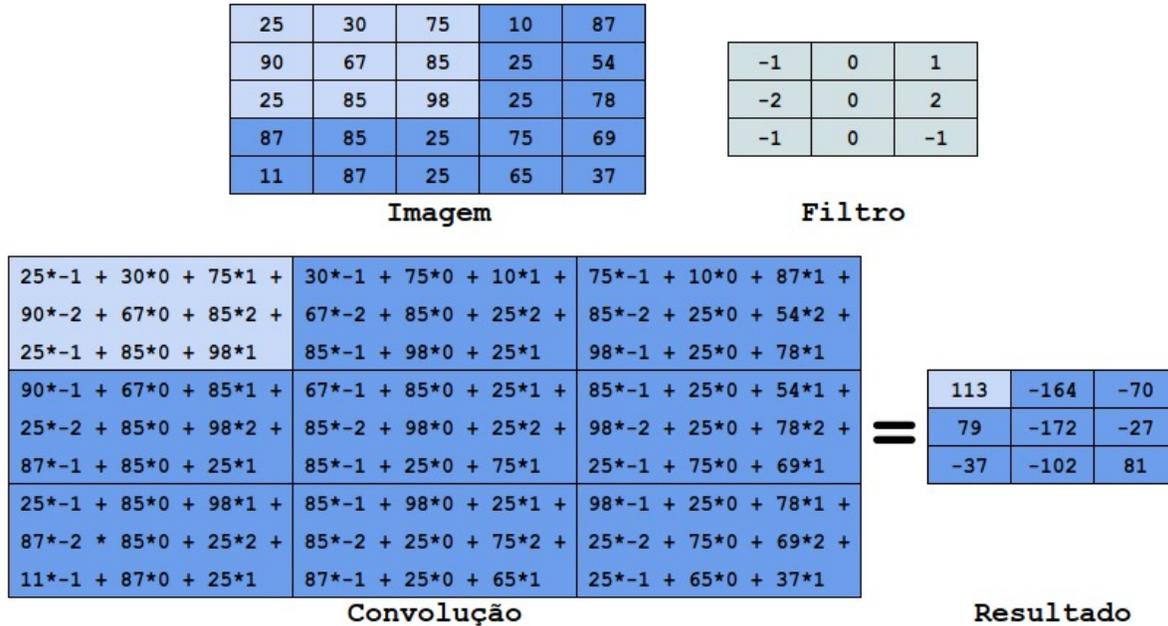


Figura 3.7: Exemplo da aplicação da convolução em uma imagem hipotética, canal único e dimensões 5x5, com um filtro de detecção de bordas horizontais de sobel, com dimensões de 3x3.

et al. (1990) desenvolveram uma CNN que foi capaz de atingir o estado da arte, em visão computacional, para diversos problemas de reconhecimento de imagens.

A rede neural convolucional proposta por LeCun et al. (1990) era organizada em dois tipos de camadas, camadas convolucionais e camadas de *downsampling*. As camadas são compostas por neurônios, os quais são associados com posições bidimensionais da imagem de entrada, juntamente com um campo receptivo.

As arquiteturas de redes neurais convolucionais atuais são divididas em estágios. Os primeiros estágios são compostos de camadas convolucionais e de *downsampling*. As camadas convolucionais consistem em mapas de atributos, os quais são conectados a cada unidade da camada anterior através de um conjunto de parâmetros, que são compartilhados entre todas as unidades. Possuem também camadas de retificação, ou funções de ativação, que são aplicadas nas saídas das camadas convolucionais. A camada ReLU (*Rectified Linear Unit*), neurônios de ativação não linear, definida pela Equação 3.4, é o tipo de camada de retificação mais comum (Shin et al., 2016; Abdel-Hamid et al., 2013).

$$f(x) = \max(0, x) \tag{3.4}$$

As camadas de *downsampling* tem como objetivo reduzir as dimensões da representação dos dados, diminuindo assim, a computação necessária para as próximas camadas. O *pooling* é uma forma de *downsampling*, sendo que o *max pooling*, técnica que computa o máximo local de uma determinada região, é o mais utilizada na literatura, devido ao fato de desprezar pequenas mudanças e distorções locais. A Figura 3.8 apresenta a aplicação de um *max pooling* 2x2, responsável por extrair o valor máximo de um grupo de 4 *pixels*, reduzindo a quantidade de *pixels* de uma imagem em 1/4.

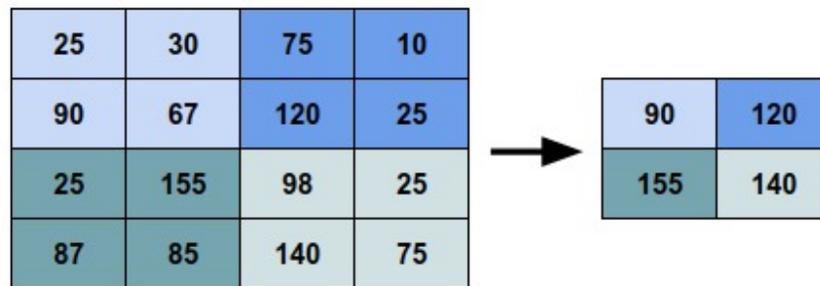


Figura 3.8: Exemplo de *max pooling*, filtro 2x2.

Alguns estágios de convolução, não-linearidade e *downsampling* são empilhados, seguidos por camadas perceptron totalmente conectadas. A Figura 3.9 apresenta uma rede convolucional genérica, composta por uma camada de entrada de 3 canais, RGB, camadas de convolução, camadas de *pooling*, e camadas perceptron totalmente conectadas.

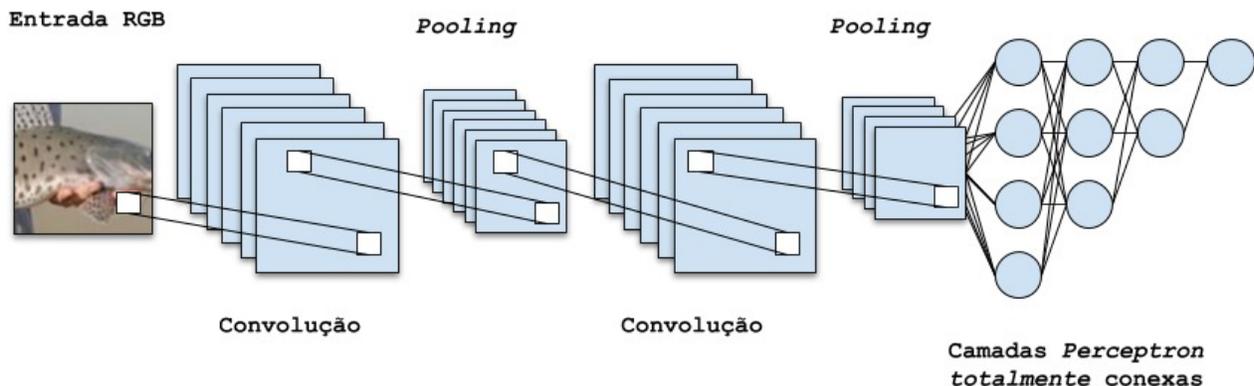


Figura 3.9: Arquitetura tradicional de um rede neural convolucional. Composta por uma camada de entrada RGB, camadas de convolução, camadas *pooling* e camadas perceptron totalmente conexas.

Atualmente podemos citar diversas arquiteturas de CNN, aplicadas à visão

computacional, sendo as principais: LeNet (LeCun et al., 1990), AlexNet (Krizhevsky et al., 2012), ZFNet (Zeiler and Fergus, 2014), GoogleNet (Szegedy et al., 2015a), Inception (Szegedy et al., 2015b), VGGNet (Simonyan and Zisserman, 2014) e ResNet (He et al., 2015). Todas venceram o desafio ImageNet em seus respectivos anos de criação (Russakovsky et al., 2015).

Embora muito empregada em problemas de classificação, as CNNs podem também ser empregadas em problemas de regressão. Hu et al. (2016) aplicaram CNN na contagem de pessoas em multidões. A solução proposta por eles consiste na utilização de uma rede treinada, de forma supervisionada, através de dois atributos da multidão, sendo eles: a contagem e a densidade da multidão. A abordagem foi testada em um conjunto de dados com 107 imagens com 45 mil humanos, sendo que em cada imagem, a multidão varia de 58 a 2201.

Yuan et al. (2014) propõe a criação de uma CNN sem a camada totalmente conectada, gerando assim uma rede puramente convolucional. Nos experimentos realizados por ele, aplicados em problemas de detecção de faces e predição de saliências em imagens, a camada de saída possui as mesmas dimensões que a camada de entrada, gerando um valor para cada pixel.

Artigo 1

**Uma investigação sobre a otimização de
parâmetros em problemas de
contagem de alevinos**

(Computer and Electronics in Agriculture)

Uma investigação sobre a otimização de parâmetros em problemas de contagem de alevinos

Adair da Silva Oliveira Junior^a, Vanessa Aparecida de Moares Weber^{b,c}, Gilberto Astolfi^{a,d}, Fabricio de Lima Weber^c, Geazy Vilharva Menezes^a, Gabriel Kirsten Menezes^a, Pedro Lucas França Albuquerque^e, Vanir Garcia^{b,d}, Eduardo Quirino Arguelho de Queiroz^b, João Victor Araújo Rozales^b, Marco Hiroshi Naka^d, Hemerson Pistori^b

^aUniversidade Federal de Mato Grosso do Sul

^bUniversidade Católica Dom Bosco

^cUniversidade Estadual de Mato Grosso do Sul

^dInstituto Federal de Mato Grosso do Sul

^eUniversity of Nebraska - Lincoln

Resumo

O objetivo deste artigo é investigar qual é a combinação de parâmetros do *software* de contagem de alevinos, que resulte em uma menor média dos erros absolutos (MAE) e na menor raiz da média dos erros ao quadrado (RMSE). Para tanto, foi criado um banco de imagens denominado FISHCV155V, o qual foi dividido em conjuntos de treino e de teste, onde foram testadas combinações dos parâmetros do *software*. A partir dos resultados obtidos, foram extraídas métricas de desempenho individual das combinações de parâmetros como MAE, média dos erros ao quadrado (MSE) e RMSE. Foram analisados quadros de vídeos comparando as combinações de parâmetros que obtiveram o melhor e o pior resultado, a fim de investigar a influência desses parâmetros no desempenho do *software*. A partir dos resultados alcançados, concluímos que a combinação com o melhor resultado apresentou 5,99 de MAE e 9,96 RMSE.

Palavras-chave: alevinos, contagem automatizada, parametrização, Pintado Real, Visão Computacional.

1. Introdução

A produção de peixe, a segunda proteína mais produzida no mundo, encontra-se em ascensão no Brasil. A previsão é de crescimento de 100% na produção de pesca e aquicultura até o ano de 2025 [1, 2]. O crescimento populacional e a busca crescente por alimentos saudáveis demandam por adoção de tecnologias e avanços na produção piscícola [3, 4].

O Brasil possui 8.501 km de costa marítima, cerca de 12% do total da água doce do mundo e mais de 2 mil espécies de peixes de água doce catalogadas, segundo Buckup et. al [5]. Este número reflete uma grande quantidade de espécies com potenciais para cultivo. Nesse sentido, a reprodução natural ou induzida de alevinos permite a sustentabilidade da atividade da piscicultura, visto que a exploração da pesca extrativista gerou desequilíbrio nas populações de peixes, resultando até em risco de extinção de algumas

espécies. Adicionalmente, as condições climáticas favorecem a produção das espécies como a tilápia, que se destaca por ser a espécie mais consumida do mundo. Os investimentos em inovação e pesquisa nos últimos anos são fatores que contribuem para o crescimento previsto [1, 6].

Apesar da piscicultura apresentar-se como uma atividade agropecuária em plena expansão pelo Brasil e no mundo [7], muitos dos seus processos demandam por tecnologias para automatizar e tornar a atividade mais eficiente, como é o caso do sistema de contagem de alevinos. O controle da regulação de estoque de alevinos é um processo fundamental para a piscicultura, porém a contagem humana e visual de alevinos é uma atividade onerosa, cansativa e passível de erros.

Equipamentos com ferramentas de software baseadas em visão computacional embutidas têm sido estudados e desenvolvidos com o intuito de apoiar esse processo de contagem [8, 9, 10, 11, 12]. Dentre esses equipamentos e softwares relatados, destacamos o contador de alevinos desenvolvido pelo grupo Inovisão, cujo principal requisito se refere a contagem de alevinos

Email address: adairsojr@gmail.com (Adair da Silva Oliveira Junior)

da espécie Pintado Real.

Destacamos também, a importância da definição e do ajuste de parâmetros que exercem influência significativa no desempenho de *softwares* em especial de contagem, bem como do relatado para contagem de alevinos. No entanto na maioria da vezes, esses parâmetros são definidos de forma *ad-hoc*. Nesse sentido, a otimização dos parâmetros fornece valores otimizados para variáveis implementadas no *software* a fim de se obter o melhor resultado nas contagens.

Neste contexto, o objetivo deste artigo é investigar, dentro de um espaço delimitado de possibilidades, qual é a combinação de parâmetros do *software* de contagem de alevinos que resulta em uma menor média dos erros absolutos e na menor raiz da média dos erros ao quadrado.

A principal contribuição deste artigo é a apresentação de uma metodologia para avaliação do desempenho do *software* de contagem de alevinos para diferentes combinações de parâmetros.

Este artigo está organizado em mais cinco seções. Na Seção 2 são apresentados trabalhos correlatos. Na Seção 3 é feita uma contextualização sobre o contador de alevinos. Na Seção 4 são apresentadas as metodologias para construção do banco de imagens e execução do experimento. Na Seção 5 são apresentados e discutidos os resultados obtidos e, por fim, na Seção 6 é apresentada a conclusão.

2. Trabalhos Correlatos

Contadores de peixes baseados em visão computacional competem com outros 3 tipos de contadores, contadores sonoros, contadores resistivos e contadores óticos [8]. Esses contadores são empregados em todas as fases da vida dos peixes, ou seja, desde a fase embrionária até a sua fase adulta.

Fan e Liu [8] utilizaram técnicas de subtração de fundo e de esqueletização em conjunto com LS-SVM (*Least Squares Support Vector Machine*) e BPNN (*Back Propagation Neural Network*) para contar peixes por vídeo. Os peixes eram dispostos em um recipiente com água e uma câmera, posicionada em cima do recipiente, dos quais se realizava a captura das imagens. Utilizaram 7 características geométricas, área, perímetro, área convexa, largura da caixa delimitadora, altura da caixa delimitadora, tamanho e número de extremidades dos esqueletos. Conseguiram uma taxa de acurácia de 98.73% sendo que a LS-SVM foi superior à BPNN.

Costa et. al [10] propuseram um sistema diferente, no qual havia a presença de duas câmeras submersas.

Essas duas câmeras realizavam a captura de imagens síncronas no momento em que os peixes, da espécie *Thunnus thynnus*, eram transportados de uma rede para gaiolas flutuantes. Eram capturados 2 quadros por segundo, os quais eram processados por um *software* baseado em redes neurais artificiais. Essas informações eram utilizadas tanto para realizar a contagem quanto para prever o comprimento dos peixes.

Duan et. al [9] desenvolveram um método, baseado em visão computacional, para realizar a contagem de ovos de peixes mortos. A contagem era baseada no princípio de que ovos mortos flutuam. Destacam que a vantagem na substituição dos métodos tradicionais de contagem por um método baseado em visão computacional é a diminuição do manejo desses ovos, ou seja, é um método menos invasivo e menos exaustivo.

Klapp et. al [12] propuseram um sistema de contagem de peixes ornamentais baseado em uma técnica híbrida, a qual utiliza sensores óticos e processamento de sinais. Destacam que com esta técnica, o ruído exercido pela oscilação da água foi diminuído.

Hernández-Ontiveros [11] desenvolveram um contador de peixes utilizando um sistema embarcado para processar as imagens, que em condições controladas de iluminação e de espécies, obteve uma acurácia de 96% na contagem, para diferentes espécies e diferentes tamanhos.

3. Contador de Alevinos

O equipamento, desenvolvido neste trabalho, que realiza a contagem de alevinos consiste de 4 componentes principais, a estrutura, a iluminação, a câmera e o *software*. A estrutura é montada de maneira que fique inclinada, em torno de 11 a 13 graus, por onde os alevinos deslizam com o auxílio da água, a qual escoar em fluxo contínuo. A inclinação e a vazão da água influenciam na velocidade em que os alevinos deslizam sobre a estrutura. Se a velocidade for alta, as imagens captadas podem não ser suficientes para realizar o rastreamento dos alevinos, por outro lado, se for baixa, o equipamento terá uma pequena taxa, alevinos por hora, de contagem.

A iluminação é também um fator importante e deve causar pouca reflexão, bem como corrigir alterações exteriores, fazendo com que as imagens sejam uniformes, otimizando o foco e o contraste. A câmera é acoplada à estrutura e realiza a captura das imagens. Os parâmetros de taxa de quadros captados por segundo e a resolução também influenciam

diretamente o sistema de contagem, uma vez que se a taxa for pequena, não haverá quadros suficientes para o rastreamento dos alevinos, em contrapartida, se a taxa for alta, acarretará em um maior consumo dos recursos do computador, podendo até impossibilitar a contagem em tempo real. A Figura 1 apresenta a estrutura do contador de alevinos.

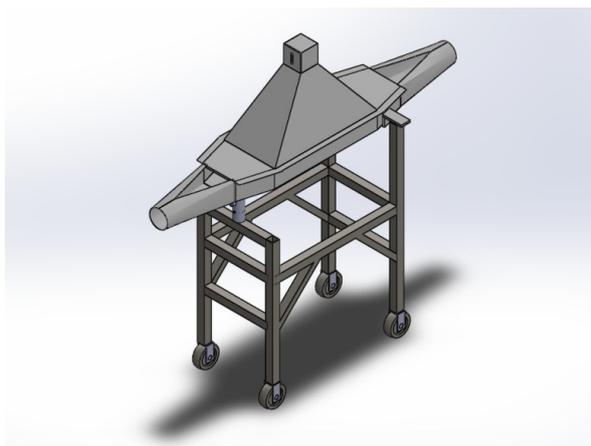


Figura 1: Protótipo atual da estrutura do contador de alevinos.

O *software* de contagem de alevinos conta com as técnicas de segmentação por subtração de fundo, detecção de componentes conexas (*Blobs*), detecção de contornos (*Find Countours*) e filtro preditivo de *Kalman*. O algoritmo de contagem funciona basicamente na segmentação por subtração de fundo e na detecção de regiões de interesse e contornos, que são rastreados através das predições realizadas pelo filtro de *Kalman*. O mesmo possui parâmetros que definem o padrão da contagem, como por exemplo a área mínima e o valor de limiarização para detecção das componentes conexas, no caso os alevinos, e desvio padrão da área dos mesmos, para correção quando há aglomerações. Em linhas gerais, o algoritmo realiza a contagem através dos seguintes passos:

1. Segmentação por subtração de fundo;
2. Detecção de regiões de interesse e contornos, para rotulação dos mesmos como candidatos a alevinos;
3. Aplicação do filtro de *Kalman* aos candidatos e associação às regiões de interesse e contornos detectados no novo quadro.
4. Verificação da quantidade de vezes que um candidato foi detectado. Se alcançar o valor de um parâmetro, denominado limiar candidato, é feita a mudança da rotulação de candidato para alevino.
5. Verificação da posição do alevino em relação à

região de contagem. Se o mesmo a ultrapassou é realizada o cômputo.

O *software* possui os seguintes parâmetros de contagem:

- Distância entre *blobs*: distância mínima entre os *blobs* de um quadro. Se um *blob* for encontrado dentro desta distância, os dois são associados como um único *blob*.
- Área do *blob*: área mínima de um *blob*. *Blobs* com áreas menores são descartados.
- Limiar *blob*: valor para segmentação por limiarização do *blob*.
- Limiar candidato: quantidade de vezes que um candidato deve ser rastreado para se tornar um alevino.
- Desvio padrão: desvio padrão. Em casos de aglomeração é feita uma análise da área do *blob* para definição se há mais de um alevino.
- Área mínima: área mínima necessária para um *blob* ser contado como um alevino. Semelhante à área do *blob*, porém *blobs* menores podem ser detectados para questões de rastreamento, sendo feita uma verificação na hora da contagem.

4. Metodologia

Nesta seção serão apresentadas as metodologias que foram empregadas na criação do banco de imagens e nos experimentos realizados neste artigo.

4.1. Criação do Banco de Imagens (FISHCV155V)

A criação do banco de imagens foi realizada no laboratório de piscicultura do INOVISAO, o qual foi montado em uma parceria entre a UCDB, IFMS e a empresa Projeto Pacu, através do equipamento de contagem de alevinos, descrito na Seção 3. Denominado FISHCV155V, o banco de imagens possui 155 vídeos, capturados por uma HD Pro WebCam C920 da Logitech, com a resolução de 640x320 *pixels* a uma taxa de 30 quadros por segundo. A quantidade de alevinos por vídeo varia de 1 até 36 unidades. As gravações foram realizadas em 3 dias, entre os meses de julho e setembro de 2018. Além da quantidade de alevinos por vídeo, o mesmo possui a massa total dos alevinos e a inclinação do equipamento, parâmetro que influencia na velocidade na qual os alevinos passam pelo mesmo. A metodologia adotada na captação dos vídeos seguiu os seguintes passos:

1. Separação dos alevinos em 2 recipientes (1) não utilizados e (2) já utilizados;
2. Separação dos alevinos que serão utilizados para construção do banco de imagens, depositados no recipiente (1);
3. Separação de um alevino do recipiente (1);
4. Captação das imagens do alevino no equipamento, associando a massa do alevino ao vídeo;
5. Deposição do alevino no recipiente (2);
6. Caso haja mais de 1 alevino no recipiente (2), captação das imagens dos alevinos no equipamento, associando a massa agregada ao vídeo;
7. Repetição do item 4 ao item 9 até que não haja mais alevinos no recipiente (1).

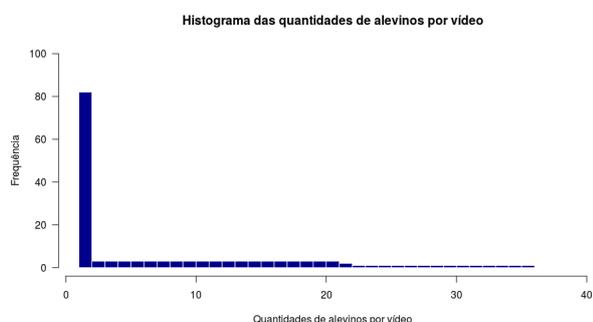


Figura 2: Histograma das quantidades de alevinos por vídeo.

Devido a disponibilidade de alevinos nos dias em que as gravações foram feitas, e da metodologia adotada na captura dos mesmos, os números de vídeos de cada quantidade de alevinos são diferentes. A Figura 2 apresenta um histograma com as frequências das quantidades de alevinos por vídeo.

4.2. Delineamento Experimental

O experimento foi realizado no banco de imagens FISHCV155V, o qual foi dividido em 2 conjuntos, treino e teste, sendo 70% e 30% o percentual de separação, respectivamente. Foram testadas combinações dos parâmetros do *software* de contagem de alevinos, com o intuito de se identificar as melhores combinações. Os parâmetros distância entre *blobs* (bd), área do *blob* (ba), limiar do *blob* (bt), limiar candidato (ct), desvio padrão (ds) e área mínima (ma) foram combinados, gerando 324 combinações diferentes, cujo os valores podem ser observados na Tabela 1.

A partir dos resultados obtidos foram extraídas algumas métricas de desempenho individual das

Tabela 1: Valores dos parâmetros do *software* contador de alevinos utilizados no experimento.

Parâmetro	Valores
Distância entre <i>blobs</i>	62, 64 e 66
Área do <i>blob</i>	33, 35 e 37
Limiar do <i>blob</i>	68 e 70
Limiar candidato	1 e 2
Desvio padrão	0.7, 0.9 e 1.1
Área mínima	1.7, 1.9 e 2.1

combinações, sendo elas: média dos erros absolutos (MAE), média dos erros ao quadrado (MSE) e a raiz da média dos erros ao quadrado (RMSE). As combinações que obtiveram o melhor resultado, ou seja, alcançaram os menores erros, foram aplicadas ao conjunto de teste. A fim de detectar as possíveis influências dos parâmetros ao *software*, foram selecionadas combinações que obtiveram os melhores e os piores resultados no treinamento, às quais foram submetidos vídeos e analisado o comportamento do algoritmo quadro a quadro.

A Figura 3 apresenta um fluxograma com a metodologia geral aplicada ao experimento de parametrização do *software* de contagem de alevinos.

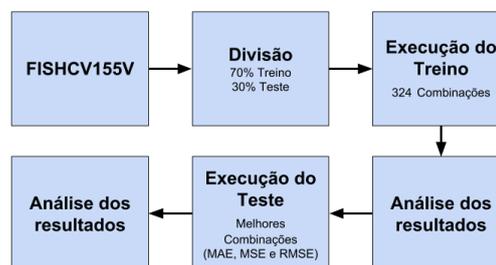


Figura 3: Fluxograma da metodologia aplicada no experimento de parametrização do *software* de contagem de alevinos.

5. Resultados e Discussão

A Tabela 2 apresenta os resultados, contendo as métricas MAE, MSE e RMSE das 20 melhores combinações de parâmetros. Podemos observar que o melhor resultado ocorre quando os parâmetros área do *blob*, limiar do *blob*, distância entre *blobs*, limiar candidato, desvio padrão e área mínima possuem os valores 62, 33, 70, 1, 1,1 e 1,7, respectivamente, sendo a média dos erros absolutos 5,99, a média dos erros ao quadrado 99,20 e raiz da média dos erros ao quadrado 9,96.

Outro fato que podemos observar na Tabela 2 é que não há mudanças nas métricas quando os valores dos parâmetros limiar do *blob* e área mínima alternam, gerando assim um empate entre 9 combinações, fazendo com que outras 8 combinações tenham o mesmo desempenho da combinação já apontada.

Tabela 2: Resultado das 20 melhores combinações de parâmetros para o conjunto de treino.

#	Parâmetros						Métricas		
	ba	bt	bd	ct	ds	ma	MAE	MSE	RMSE
1	62	33	70	1	1,1	1,7	5,99	99,20	9,96
2	62	33	70	1	1,1	1,9	5,99	99,20	9,96
3	62	33	70	1	1,1	2,1	5,99	99,20	9,96
4	62	35	70	1	1,1	1,7	5,99	99,20	9,96
5	62	35	70	1	1,1	1,9	5,99	99,20	9,96
6	62	35	70	1	1,1	2,1	5,99	99,20	9,96
7	62	37	70	1	1,1	1,7	5,99	99,20	9,96
8	62	37	70	1	1,1	1,9	5,99	99,20	9,96
9	62	37	70	1	1,1	2,1	5,99	99,20	9,96
10	64	33	70	1	1,1	1,7	6,01	100,03	10,00
11	64	33	70	1	1,1	1,9	6,01	100,03	10,00
12	64	33	70	1	1,1	2,1	6,01	100,03	10,00
13	64	35	70	1	1,1	1,7	6,01	100,03	10,00
14	64	35	70	1	1,1	1,9	6,01	100,03	10,00
15	64	35	70	1	1,1	2,1	6,01	100,03	10,00
16	64	37	70	1	1,1	1,7	6,01	100,03	10,00
17	64	37	70	1	1,1	1,9	6,01	100,03	10,00
18	64	37	70	1	1,1	2,1	6,01	100,03	10,00
19	62	33	68	1	1,1	1,7	6,02	100,50	10,02
20	62	33	68	1	1,1	1,9	6,02	100,50	10,02

A Tabela 3 apresenta os resultados, contendo as métricas MAE, MSE e RMSE das 20 piores combinações de parâmetros. Podemos observar que o pior resultado ocorre quando os parâmetros área do *blob*, limiar do *blob*, distância entre *blobs*, limiar candidato, desvio padrão e área mínima possuem os valores 66, 37, 68, 1 0,7 e 2,1, respectivamente, sendo a média dos erros absolutos 6,96, a média dos erros ao quadrado 119,81 e raiz da média dos erros ao quadrado 10,95.

A Tabela 4 apresenta as métricas, média dos erros absolutos, média dos erros ao quadrado e raiz da média dos erros ao quadrado, das 9 combinações, que obtiveram o melhor resultado no conjunto de treino, para o conjunto de teste.

A seguir foi uma análise de trechos, quadro a quadro, de 2 vídeos para os parâmetros que obtiveram o melhor e o pior resultado, combinação de posição 1, observada na Tabela 2, e a combinação de posição 324, observada na Tabela 3, com o objetivo de se identificar os possíveis erros do algoritmo de contagem. Estão apresentadas figuras contendo o rastreamento dos alevinos, nas quais serão apresentadas também as informações referentes a contagem, representada pelo valor "C", e ao quadro, representado pelo valor "F", atuais no canto superior direito. Os rastros vermelhos indicam as predições realizadas pelo filtro de Kalman,

Tabela 3: Resultado das 20 piores combinações de parâmetros para o conjunto de treino.

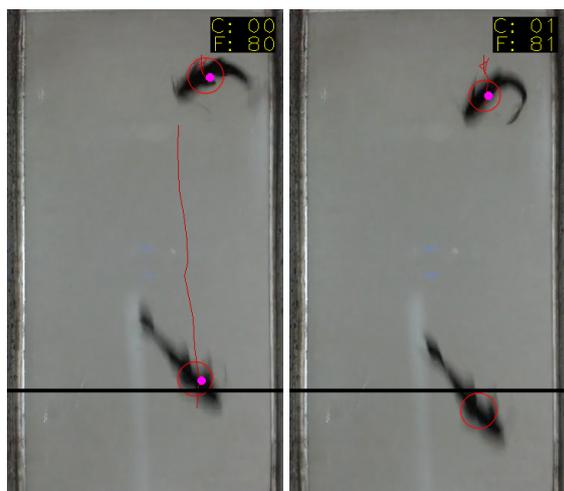
#	Parâmetros						Métricas		
	ba	bt	bd	ct	ds	ma	MAE	MSE	RMSE
305	66	37	68	2	0,7	1,7	6,94	116,23	10,78
306	66	37	68	2	0,7	1,9	6,94	116,23	10,78
307	66	37	68	2	0,7	2,1	6,94	116,23	10,78
308	62	33	68	1	0,7	1,9	6,95	119,60	10,94
309	62	33	68	1	0,7	2,1	6,95	119,60	10,94
310	62	35	68	1	0,7	1,7	6,95	119,60	10,94
311	62	35	68	1	0,7	1,9	6,95	119,60	10,94
312	62	35	68	1	0,7	2,1	6,95	119,60	10,94
313	62	37	68	1	0,7	1,7	6,95	119,60	10,94
314	62	37	68	1	0,7	1,9	6,95	119,60	10,94
315	62	37	68	1	0,7	2,1	6,95	119,60	10,94
316	66	33	68	1	0,7	1,7	6,96	119,81	10,95
317	66	33	68	1	0,7	1,9	6,96	119,81	10,95
318	66	33	68	1	0,7	2,1	6,96	119,81	10,95
319	66	35	68	1	0,7	1,7	6,96	119,81	10,95
320	66	35	68	1	0,7	1,9	6,96	119,81	10,95
321	66	35	68	1	0,7	2,1	6,96	119,81	10,95
322	66	37	68	1	0,7	1,7	6,96	119,81	10,95
323	66	37	68	1	0,7	1,9	6,96	119,81	10,95
324	66	37	68	1	0,7	2,1	6,96	119,81	10,95

Tabela 4: Métricas das melhores combinações de parâmetros para o conjunto de teste.

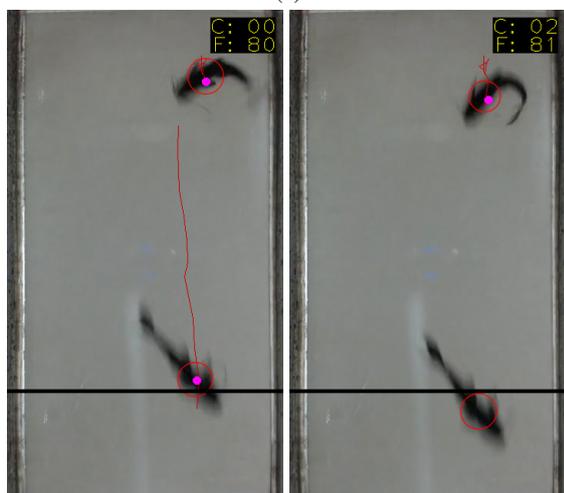
	Parâmetros					Métricas		
	ba	bt	bd	ct	ds	ma	MAE	MSE
62	33	70	1	1,1	1,7	5,00	83,00	9,11
62	33	70	1	1,1	1,9	5,00	83,00	9,11
62	33	70	1	1,1	2,1	5,00	83,00	9,11
62	35	70	1	1,1	1,7	5,00	83,00	9,11
62	35	70	1	1,1	1,9	5,00	83,00	9,11
62	35	70	1	1,1	2,1	5,00	83,00	9,11
62	37	70	1	1,1	1,7	5,00	83,00	9,11
62	37	70	1	1,1	1,9	5,00	83,00	9,11
62	37	70	1	1,1	2,1	5,00	83,00	9,11

os círculos sem preenchimento com bordas vermelhas representam os *blobs*. Os círculos com os centros rosas indicam alevinos e os com centros brancos indicam candidatos a alevinos.

No vídeo 29, entre os quadros 80 e 81, o algoritmo, configurado com a combinação de pior resultado, realizou a contagem de três alevinos, quando na verdade era para se contar apenas um, como podemos observar na Figura 4b. A Figura 4a apresenta os mesmos quadros do vídeo 29, porém com o algoritmo configurado na combinação de melhor resultado, onde o algoritmo realizou a contagem correta. Esse é um problema de falso-positivo. Como podemos observar nas Tabelas 2 e 3 as combinações com o melhor e o pior resultado possuem valores diferentes para o parâmetro desvio padrão. Este parâmetro define o limite superior e inferior que a área do *blob* deve possuir para não ser considerado um peixe, muito pequeno, um peixe ou um aglomerado de peixe. Como o desvio padrão é menor para a combinação de pior resultado em relação à combinação de melhor resultado, 0,7 e 1,1, respectivamente, o algoritmo interpreta a área do *blob*



(a)

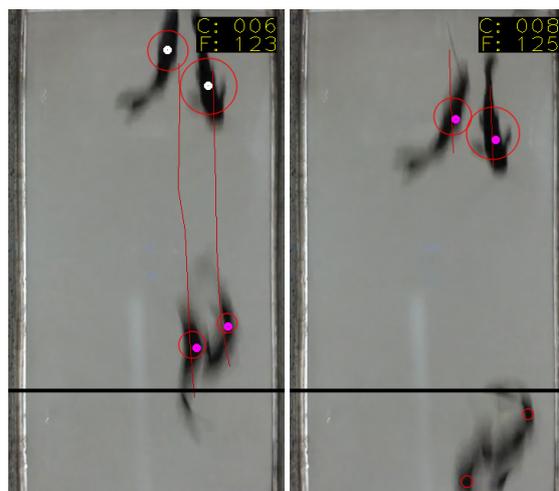


(b)

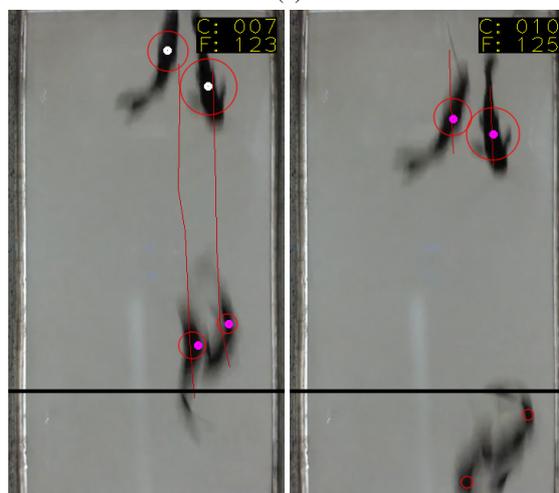
Figura 4: Comparação da contagem realizada no vídeo 29, nos quadros 80 e 81, entre as combinações de parâmetros que obtiveram o melhor e o pior resultado, onde a pior combinação realizou a contagem de um falso-positivo.

como um aglomerado e realiza a contagem de dois alevinos ao invés de um.

No vídeo 43, entre os quadros 123 e 125, o algoritmo, configurado com a combinação de pior resultado, realizou a contagem de três alevinos, quando na verdade era para se contar apenas dois, como pode ser observado na Figura 5. Esse é um problema de falso-positivo, e uma das possíveis causas é a relação dos parâmetros de área do *blob*, desvio padrão e área mínima, que para esses casos, quando não há aglomeração, esses valores estão maiores do que o ideal, fazendo o algoritmo realizar a contagem de um número maior de alevinos



(a)



(b)

Figura 5: Comparação da contagem realizada no vídeo 43, nos quadros 123 e 125, entre as combinações de parâmetros que obtiveram o melhor e o pior resultado, onde a pior combinação realizou a contagem de um falso-positivo. Há diferença na contagem anterior ao quadro 123 devido a erros de contagens nos quadros anteriores.

do que realmente estão passando pelo equipamento.

6. Conclusão

A partir dos resultados alcançados, concluímos que a combinação de parâmetros do *software* de contagem de alevinos, área do *blob*, limiar do *blob*, distância entre *blobs*, limiar candidato, desvio padrão e área mínima, com os valores 62, 33, 70, 1 1,1 e 1,7, respectivamente, apresentou a menor MAE, 5,99, e a menor RMSE, 9,96, perfazendo assim, a melhor combinação de parâmetros das 324 combinações de

parâmetros do *software* testadas, para o banco de imagens FISHCV155V.

Agradecimentos

Este trabalho recebe apoio da Universidade Católica Dom Bosco, UCDB, do Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPQ, da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, CAPES, e da Fundação de Apoio ao Desenvolvimento do Ensino, Ciência e Tecnologia do Estado de Mato Grosso do Sul, Fundect, através do termo de outorga 131/2016.

Referências

- [1] FAO, Novo relatório da fao aponta que produção da pesca e aquicultura no brasil deve crescer mais de 100% até 2025, <http://www.fao.org/brasil/noticias/detail-events/pt/c/423722/> Acessado em: 01/04/2017 (2016).
- [2] X. Zhou, Notes from the aquaculture statistician, FAO Aquaculture Newsletter 1 (58) (2018) 6–8.
- [3] X. Zhou, An overview of recently published global aquaculture statistics, FAO Aquaculture Newsletter 1 (56) (2017) 6.
- [4] M. F. Brabo, L. F. S. Pereira, J. V. M. Santana, D. A. V. Campelo, G. C. Veras, Cenário atual da produção de pescado no mundo, no brasil e no estado do pará: ênfase na aquicultura/current scenario of fish production in the world, brazil and pará state: emphasis on aquaculture, Acta of Fisheries and Aquatic Resources 4 (2) (2016) 50–58.
- [5] P. A. Backup, N. A. Menezes, M. S. Ghazzi, Catálogo das espécies de peixes de água doce do Brasil, Vol. 1, Museu Nacional Rio de Janeiro, 2007.
- [6] L. Sidonio, I. Cavalcanti, L. Capanema, R. Morch, G. Magalhães, J. Lima, V. Burns, A. J. Alves Júnior, R. Mungjoli, Panorama da aquicultura no brasil: desafios e oportunidades, BNDES setorial 35 (2012) 421–463.
- [7] FAO, Pma e oms (2017) o estado da segurança alimentar e nutricional no mundo 2017: Construindo resiliência para a paz e segurança alimentar, FAO.
- [8] L. Fan, Y. Liu, Automate fry counting using computer vision and multi-class least squares support vector machine, Aquaculture 380 (2013) 91–98.
- [9] Y. Duan, L. H. Stien, A. Thorsen, Ø. Karlsen, N. Sandlund, D. Li, Z. Fu, S. Meier, An automatic counting system for transparent pelagic fish eggs based on computer vision, Aquacultural engineering 67 (2015) 8–13.
- [10] C. Costa, M. Scardi, V. Vitalini, S. Cataudella, A dual camera system for counting and sizing northern bluefin tuna (*thunnus thynnus*; linnaeus, 1758) stock, during transfer to aquaculture cages, with a semi automatic artificial neural network tool, Aquaculture 291 (3-4) (2009) 161–167.
- [11] J. Hernández-Ontiveros, E. Inzunza-González, E. García-Guerrero, O. López-Bonilla, S. Infante-Prieto, J. R. Cárdenas-Valdez, E. Tlelo-Cuautle, Development and implementation of a fish counter by using an embedded system, Computers and Electronics in Agriculture 145 (2018) 53–62.
- [12] I. Klapp, O. Arad, L. Rosenfeld, A. Barki, B. Shaked, B. Zion, Ornamental fish counting by non-imaging optical system for real-time applications, Computers and Electronics in Agriculture 153 (2018) 126–133.

Artigo 2

**Estimativa da massa de alevinos:
um comparativo entre redes
convolucionais profundas
e algoritmos rasos**

(Aquacultural Engineering)

Estimativa da massa de alevinos: um comparativo entre redes convolucionais profundas e algoritmos rasos

Adair da Silva Oliveira Junior^a, Vanessa Aparecida de Moares Weber^{b,c}, Gilberto Astolfi^{a,d}, Fabricio de Lima Weber^c, Geazy Vilharva Menezes^a, Gabriel Kirsten Menezes^a, Celso Soares Costa^{b,d}, Diego André Sant'Ana^{b,d}, Marcio Carneiro Brito Pache^{b,d}, Marco Hiroshi Naka^d, Hemerson Pistori^b

^aUniversidade Federal de Mato Grosso do Sul

^bUniversidade Católica Dom Bosco

^cUniversidade Estadual de Mato Grosso do Sul

^dInstituto Federal de Mato Grosso do Sul

Resumo

O objetivo deste artigo é estimar a massa de alevinos da espécie Pintado Real a partir de imagens, por meio de técnicas de classificação e regressão implementadas com algoritmos rasos e profundos a fim de apoiar o processo de produção de peixes. Para tanto, foi criado um banco de imagens denominado FISHCV1206FSEG, o qual é composto por 1206 imagens de alevinos com suas respectivas massas anotadas. Foram extraídas características, área e perímetro, dos contornos dos peixes as quais foram submetidas aos algoritmos de classificação e regressão J48, SMO e IBK e ao algoritmo de regressão linear. As imagens foram submetidas também às redes neurais convolucionais ResNet50, InceptionV3, Exception, VGG16 e VGG19. Como resultado obtivemos um modelo de regressão linear capaz de prever a massa de um alevino de Pintado Real com erro médio de 1.5 grama e uma diferença positiva de 9% da rede neural convolucional ResNet50, que obteve uma acurácia de 67.08% no conjunto teste, em relação ao melhor resultado dos algoritmos rasos, J48 que obteve uma acurácia de 58.2%.

Palavras-chave: alevinos, aprendizado profundo, classificação, estimativa da massa, Pintado Real, visão computacional.

1. Introdução

A busca por alimentos mais saudáveis impulsionam o crescimento da produção piscícola. Segundo a Organização das Nações Unidas para Alimentação e Agricultura (FAO) [1], o peixe é a segunda proteína animal mais produzida no mundo [2, 3]. Há mais de 22.000 espécies de peixes catalogadas, segundo Backup [4] no Brasil são mais de 2.000 espécies de água doce, trazendo à piscicultura uma grande quantidade de potenciais espécies para cultivo [5]. Estima-se que o Brasil terá um crescimento de 100% em sua produção de pesca e aquicultura até o ano de 2025 [6]. Além das condições climáticas, o Brasil possui cerca de 12% do total da água doce do mundo, segundo dados da Embrapa, dados esses que apresentam o país como favorável às atividades de pesca e piscicultura. Afim

de apoiar e garantir essas atividades, investimentos têm sido feitos em pesquisa de desenvolvimento tecnológico [7, 8].

Devido aos fatos e dados apontados, é consequente a necessidade de desenvolvimento tecnológico para dar suporte à produção de peixes, principalmente à piscicultura. Essas inovações, além de sustentar o aumento da produção, podem apoiar a atividade de forma a permitir a garantia da qualidade dos produtos. Dentro dessa perspectiva por inovações, destacamos que os estudos como contagem de espécimes [9, 10, 11, 12, 13], medição biométrica [14, 15, 16], estimativa da massa [17, 14, 18, 19, 20, 21, 22, 23, 24, 25], relação entre dimensões e massa [26], classificação para identificação de espécies [27, 28], identificação de sexo, avaliação da qualidade das cores de peixes ornamentais e monitoramento do bem-estar que tem sido apoiados por técnicas de visão computacional [24]. Isso se deve ao fato de que a visão computacional tem como objetivo, o emprego de técnicas computacionais

Email address: adairsojr@gmail.com (Adair da Silva Oliveira Junior)

para a extração de informações úteis a partir de imagens [29], sendo assim, constitui um método não invasivo, o que reduz o estresse dos peixes. A fim de complementar as técnicas de visão computacional, a aprendizagem de máquina, que é uma subárea da inteligência artificial [30] e tem como premissa resolver diversos problemas de classificação e regressão [31], por meio de aprendizado através de exemplos, vem contribuindo para a construção de ferramentas para piscicultura.

A zootecnia é a área da ciência que se preocupa com o bem estar e produtividade de animais e consequentemente, da atividade da piscicultura [32]. Para garantir a qualidade nutricional e decidir sobre as melhores dietas disponibilizadas para os peixes e assim preservar a saúde dos animais, é importante conhecer informações sobre a massa dos mesmos durante seus estágios de crescimento. Essas informações também podem apoiar produtores de modo a facilitar decisões sobre a classificação, e definição da melhor época para colheita e comercialização dos peixes. Essa tarefa quando exercida exclusivamente por visão humana é cansativa e propícia a erros. Consequente estimar a massa de peixe por meio de técnicas de visão computacional é uma forma de amparar tecnologicamente a atividade da piscicultura.

Muitos autores criaram modelos para estimativa da massa para outros animais de interesse da zootecnia, tais como bovinos [33, 34, 35], suínos [36], ovelhas [37] e até mesmo peixes. Sendo que na sua maioria, essas pesquisas estão amparadas em técnicas de regressão para apresentar como resultado, o peso efetivo do animal estudado. Porém, na piscicultura, na fase do início da engorda, alguns animais apresentam diferença de peso com relação à outros e os produtores precisa separá-los para que estes se recuperem. Sendo assim, esse procedimento é constituído basicamente por uma classificação dos animais por sua massa. Nesse sentido, observa-se a necessidade de apoio computacional para, por meio de imagens, classificar a massa de alevinos.

Sendo assim, o objetivo deste artigo é estimar a massa de alevinos da espécie Pintado Real a partir de imagens, por meio de técnicas de classificação e regressão implementadas com algoritmos rasos e profundos, a fim de apoiar o processo de produção de peixes. Podemos destacar as contribuições deste trabalho como: apresentar uma metodologia para classificação da massa de alevinos de forma não invasiva; analisar e comparar resultados de diferentes algoritmos rasos, de classificação e de regressão, e finalmente, compará-los com resultados de técnicas de classificação profunda adaptadas para teste com

transferência de aprendizado e ajuste fino.

Este artigo está organizado em mais cinco seções. Na Seção 2 são apresentados trabalhos correlatos. Na Seção 3 é feita uma contextualização sobre o contador de alevinos, o qual é utilizado na captação das imagens. Na Seção 4 são apresentadas as metodologias para construção do banco de imagens e execução dos experimentos. Na Seção 5 são apresentados e discutidos os resultados obtidos e, por fim, na Seção 6 é apresentada a conclusão.

2. Trabalhos Correlatos

A estimativa da massa, apoiada por técnicas de visão computacional, vem sendo explorada para diversos tipos de animais [18, 19, 20, 21, 22, 23, 24, 25]. Zion [24] realizou uma revisão de como a visão computacional tem sido explorada para auxiliar a piscicultura, não só na estimativa da massa de peixes, mas também na contagem, medição, identificação do sexo, avaliação da qualidade de cores em peixes ornamentais, identificação de espécies e monitoramento do bem-estar.

Viazzi et. al [25] estimaram a massa de peixes da espécie Jade Perch (*Scortum Barcoo*) dentro de tanques, de modo a minimizar o estresse e dano causado nos mesmos. Essa espécie é encontrada nos rios da Austrália. Foram realizados experimentos de regressão para criação de modelos, os quais correlacionam as formas do peixe com a sua massa. Através de um banco com 120 imagens, foi realizada a regressão utilizando-se a área do peixe, desconsiderando a cauda, e obtiveram erros entre 3 e 9%.

Lines et. al [23] realizaram experimentos com câmeras estereoscópicas para estimar a massa de peixes da espécie salmão. Os mesmos usaram 5 dimensões lineares extraídas de diversas regiões do peixe e obtiveram modelos matemáticos de regressão com erros menores que 0.5%. A extração dessas medidas através das imagens são realizadas com erros menores que 10%. Os maiores problemas apontados foram a complexidade das imagens obtidas e a identificação dos modelos morfológicos que variam de acordo com a maturidade, tensão e cultura do salmão. Esses experimentos foram realizados com 60 imagens de 17 peixes com os pesos entre as faixas de 0.7 à 5.7 quilogramas.

Odone et. al [22] propuseram a criação de um sistema de visão computacional treinável, através de máquinas de vetores de suporte, para estimativa da massa de peixes a partir de medidas extraídas das imagens. Diferente da visão estereoscópica utilizada por Lines et. al [23], as imagens são capturadas por

duas câmeras posicionadas em cima e na lateral de um tubo retangular transparente, por onde os peixes passam nadando. As medidas extraídas das imagens capturadas pela câmera na lateral são: área, perímetro, comprimento, a relação área/comprimento, a relação área/perímetro, a menor e a maior largura do peixe. A câmera superior captura as mesmas medidas, exceto a largura mínima que causa instabilidade, pois depende da pose do peixe no momento em que a imagem é capturada. Nos experimentos foram usados 25 pares de entradas e saídas para treinamento e 25 entradas para teste, onde o sistema foi capaz de predizer a massa com um erro médio de 4%.

Puig-Pons et. al [17] combinaram técnicas acústicas e visão computacional para o desenvolvimento de um procedimento automático capaz de estimar a massa viva de peixes da *Thunnus thynnus* durante os processos de transferências. Um sonar de 200 kHz e uma câmera estéreo foram combinadas na captura de imagens. O procedimento desenvolvido foi capaz de realizar a estimativa da massa com um erro médio de 10%.

3. Hardware de Captura de Imagens

O equipamento que realiza a contagem de alevinos consiste de 4 componentes principais sendo eles: a estrutura, a iluminação, a câmera e o *software*. A estrutura é montada de maneira que fique inclinada, em torno de 11 a 13 graus, por onde os alevinos deslizam com o auxílio da água, a qual escoar em fluxo contínuo. A inclinação e a vazão da água influenciam na velocidade em que os alevinos deslizam sobre a estrutura. Se a velocidade for alta, as imagens captadas podem não ser suficientes para realizar o rastreamento dos alevinos, por outro lado, se for baixa, o equipamento terá uma pequena taxa, alevinos por hora, de contagem.

A iluminação é também um fator importante e deve causar pouca reflexão, bem como corrigir alterações exteriores, fazendo com que as imagens sejam uniformes, otimizando o foco e o contraste. A câmera é acoplada à estrutura e realiza a captura das imagens. Os parâmetros de taxa de quadros captados por segundo e a resolução também influenciam diretamente o sistema de contagem, uma vez que se a taxa for pequena, não haverá quadros suficientes para o rastreamento dos alevinos, em contra partida, se a taxa for alta, acarretará em um maior consumo dos recursos do computador, podendo até impossibilitar a contagem em tempo real.

O *software* opera em um computador, atualmente em um *laptop* por razões de mobilidade, responsável pelo processamento das imagens, e portanto, da contagem. A

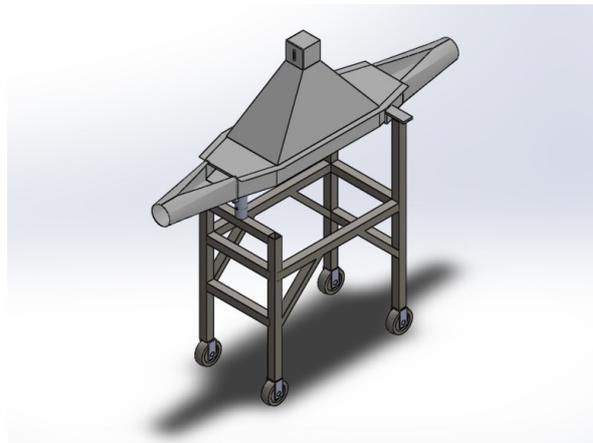


Figura 1: Hardware de captura de imagens.

Figura 1 apresenta a estrutura do Hardware de captura de imagens.

O *software* de contagem de alevinos conta com as técnicas de segmentação por subtração de fundo (*Background Subtraction*), detecção de pontos de interesse (*Blob Detection*), detecção de contornos (*Find Countours*) e filtro preditivo de *Kalman* (*Kalman Filter*). O algoritmo de contagem funciona basicamente na segmentação por subtração de fundo e na detecção de regiões de interesse e contornos, que são rastreados através das predições realizadas pelo filtro de *Kalman*.

4. Metodologia

Nesta seção serão apresentadas as metodologias que foram empregadas na criação do banco de imagens e nos experimentos realizados neste artigo.

4.1. Criação do Banco de Imagens (FISHCV1206FSEG)

A criação do banco de imagens foi realizada no laboratório de piscicultura do INOVISAO, o qual foi montado em uma parceria entre a UCDB, IFMS e a empresa Projeto Pacu, através do equipamento de contagem de alevinos, descrito na Seção 3. Primeiramente foi criado um banco de imagens denominado FISHCV155V, o qual possui 155 vídeos, capturados por uma HD Pro WebCam C920 da Logitech, com a resolução de 640x320 *pixels* a uma taxa de 30 quadros por segundo. A quantidade de alevinos por vídeo varia de 1 até 36 unidades. As gravações foram realizadas em 3 dias, entre os meses de julho e setembro de 2018. Além da quantidade de alevinos por vídeo, o mesmo possui a massa total dos alevinos e a

inclinação do equipamento, parâmetro que influencia na velocidade na qual os alevinos passam pelo mesmo. A metodologia adotada na captação dos vídeos seguiu os seguintes passos:

1. Separação dos alevinos em 2 recipientes (1) não utilizados e (2) já utilizados;
2. Separação dos alevinos que serão utilizados para construção do banco, depositados no recipiente (1);
3. Disposição de um recipiente com água na balança de precisão;
4. Realização da função tara;
5. Separação de um alevino do recipiente (1);
6. Realização do aferimento da massa do alevino;
7. Captação das imagens do alevino no equipamento, associando a massa do alevino ao vídeo;
8. Deposição do alevino no recipiente (2);
9. Caso haja mais de 1 alevino no recipiente (2), captação das imagens dos alevinos no equipamento, associando a massa agregada ao vídeo;
10. Repetição do item 4 ao item 9 até que não haja mais alevinos no recipiente (1).

O banco de imagens FISHCV1206FSEG possui 1206 segmentos, extraídos dos 79 vídeos com apenas um alevino do banco de imagens FISHCV155V, portanto, o banco conta com 79 alevinos diferentes, os quais possuem massas que variam entre 6,45 a 18,51 gramas. Essa metodologia foi adotada devido a rastreabilidade dos alevinos, pois nos vídeos com apenas um alevino é possível associar a massa total do vídeo ao próprio alevino. Esses segmentos são recortes de 200x200 *pixels* com os alevinos no centro dos mesmos. Foram recortados todos os quadros onde os alevinos aparecem. A Figura 2 apresenta exemplos de segmento do banco de imagens FISHCV1206FSEG.



Figura 2: Exemplos do banco de imagens FISHCV1206FSEG.

4.2. Experimento de classificação e regressão da massa de alevinos utilizando algoritmos rasos

O experimento foi realizado através das imagens do banco FISHCV1206FSEG, das quais foram extraídas características, área e perímetro em *pixels*, dos contornos dos alevinos. A Figura mostra a detecção das bordas dos alevinos das quais são extraídas as características. O experimento foi realizado em duas abordagens, de classificação e de regressão. Na abordagem de classificação os alevinos foram separados em 3 classes, sendo elas: abaixo de 12 gramas, entre 12 e 15 gramas e acima de 15 gramas.

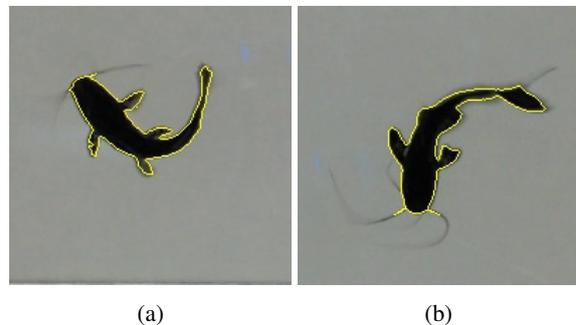


Figura 3: Exemplos da extração das características dos contornos dos alevinos. O alevino da Figura 3a possui 2537 *pixels* de área e 474,22 *pixels* de perímetro e o alevino da Figura 3b possui 2747,5 *pixels* de área e 472,86 *pixels* de perímetro.

As faixas foram definidas de maneira a proporcionar um equilíbrio de espécimes para cada classe, porém como cada alevino obteve quantidades diferentes de quadros nos vídeos, as classes ficaram desbalanceadas em relação ao número de imagens. A Figura 4 apresenta a distribuição dos dados para o problema de classificação.

Foram utilizados os seguintes algoritmos de classificação: IBK, com $k = 3$, J48, e SMO com a técnica de validação cruzada (k folds) com 10 dobras, como é indicado no trabalho de [38]. Os algoritmos de classificação foram avaliados de acordo com as seguintes métricas: acurácia, revocação e medida-F.

Na abordagem de regressão, foram utilizados os mesmo algoritmos com a adição do algoritmo de regressão linear. Os algoritmos de regressão foram avaliados de acordo com as seguintes métricas: coeficiente de correlação (CC), média dos erros absolutos (MAE) e a raiz da média dos erros ao quadrado (RMSE).

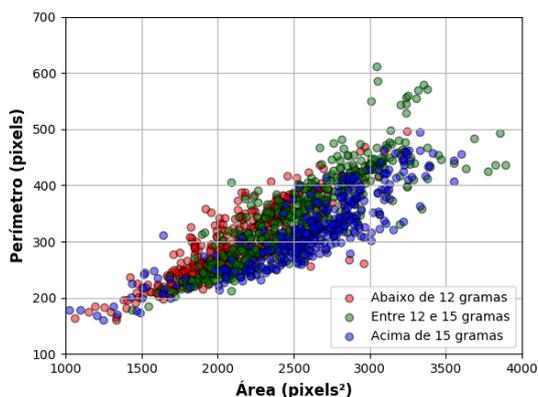


Figura 4: Distribuição dos dados do banco de imagens FISHCV1206FSEG.

4.3. Experimento de classificação da massa de alevinos utilizando aprendizagem profunda

Para a realização deste experimento, foi utilizado o banco de imagens FISHCV1206FSEG, o qual foi dividido em 3 conjuntos: treino, validação e teste, sendo 60%, 20% e 20%, respectivamente, os percentuais de separação. Foram comparadas 5 arquiteturas de redes neurais convolucionais, sendo elas: InceptionV3, ResNet50, VGG16, VGG19 e Xception. As arquiteturas foram avaliadas de acordo com a acurácia no treino, na validação e no teste.

Além das diferentes arquiteturas, foram comparadas diferentes abordagens de treinamento, sendo elas: sem transferência de aprendizagem, com transferência de aprendizado e ajuste fino. A técnica sem transferência de aprendizagem é o método tradicional de treinamento de uma rede neural convolucional, onde os pesos da rede iniciam com valores aleatórios. A transferência de aprendizagem é um técnica que aproveita o conhecimento aprendido em outros problemas, portanto a rede não é inicializada com os pesos aleatórios, mas sim com o pesos importados de outro problema. Neste trabalho, os pesos foram importados do desafio IMAGENET [39]. A técnica de ajuste fino permite, ou não, o retreinamento dos pesos importados pela técnica transferência de aprendizagem, a qual possui um parâmetro que define qual porcentagem da rede será retreinada. Foram testadas as porcentagens 25%, 50%, 75% e 100% para todas as arquiteturas. A Figura 5 apresenta um fluxograma da metodologia empregada nos experimentos deste artigo.

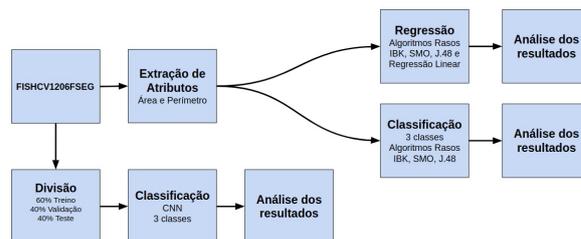


Figura 5: Metodologia empregada nos experimentos.

5. Resultados e Discussão

A Tabela 1 apresenta os desempenhos dos algoritmos rasos utilizados na abordagem de classificação. Como destacado, o algoritmo J48 obteve o melhor resultado, atingindo uma acurácia de 0.582, ou 58.2%, revocação 0.585 e medida-F 0.583. Um dos possíveis motivos para o bom desempenho do J48, em relação aos outros utilizados, é pela forma que os dados ficaram distribuídos.

Tabela 1: Resultados do experimento de classificação por algoritmos rasos.

Algoritmo de Classificação	Métrica		
	Acurácia	Revocação	Medida-F
J48	0.582	0.585	0.583
SMO	0.568	0.528	0.508
IBK (k = 3)	0.560	0.561	0.559

Como podemos ver na Figura 4 algumas amostras, de diferentes classes, se misturam, principalmente se compararmos a classe dentro do peso com as demais, motivos que diminuam o desempenho dos algoritmos de classificação.

As Figuras 6, 7 e 8 representam as matrizes de confusão dos algoritmos J48, SMO e IBK, respectivamente. Como podemos ver, as matrizes de confusão são bem próximas umas das outras. Podemos observar que os erros de predição se concentram em torno da diagonal principal.

Como podemos observar na matriz de confusão do algoritmo J48, o maior número de erros ocorreu quando o algoritmo classificou como entre 12 e 15 gramas alevinos que pertencem à classe abaixo de 12 gramas, sendo 124 esta quantidade. Um número bem próximo, 121, de erros aconteceu na situação inversa, evidenciando uma semelhança maior entre essas duas classes. Já na matriz de confusão do algoritmo SMO ficou evidente que a maior quantidade de erros ocorreu quando o algoritmo classificou como abaixo de 12 gramas alevinos que pertencem à classe

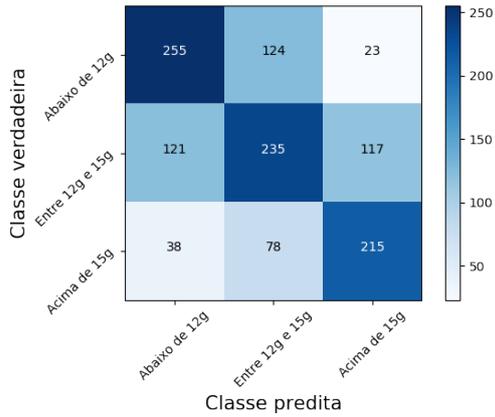


Figura 6: Matriz de confusão do algoritmo J48.

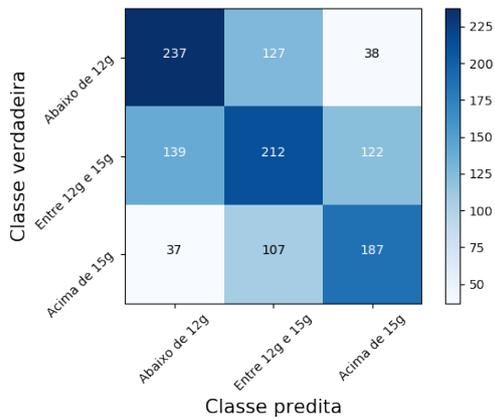


Figura 7: Matriz de confusão do algoritmo SMO.

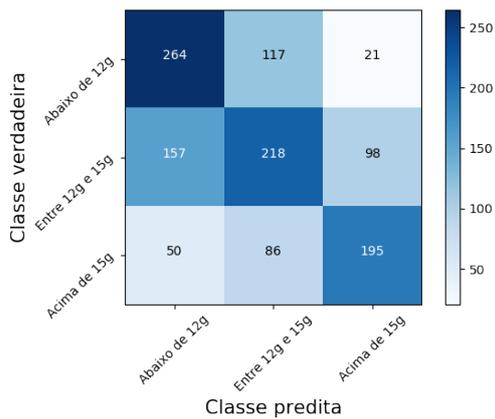


Figura 8: Matriz de confusão do algoritmo IBK ($k = 3$).

entre 12 e 15 gramas, sendo 139 esta quantidade. Na matriz de confusão do algoritmo IBK ocorreu o mesmo

que o algoritmo SMO, os erros se concentraram na classificação como abaixo de 12 gramas alevinos que pertencem à classe entre 12 e 15 gramas, sendo 157 esta quantidade.

A Tabela 2 apresenta os desempenhos dos algoritmos utilizados na abordagem de regressão. Como destacados, os algoritmos regressão linear e SMO obtiveram os melhores resultados, sendo 0.005 a diferença do coeficiente de correlação entre esses dois algoritmos. Se analisarmos o tempo de processamento e a pequena diferença de desempenho entre esses dois algoritmos, podemos considerar a regressão linear como o melhor resultado.

Tabela 2: Resultados do experimento de regressão por algoritmos rasos.

Algoritmo de Regressão	Métrica		
	CC	MAE	RMSE
Regressão Linear	0.5876	1.5127	1.9572
IBK ($k = 3$)	0.6009	1.5047	1.9606
SMO	0.5871	1.5401	1.9238
J48	0.5559	1.5869	2.0649

O algoritmo SMO possui ainda outra desvantagem perante a regressão linear, a de necessitar normalizar os dados no treinamento, no teste e posteriormente na utilização do modelo aprendido. Portanto, na utilização do modelo, os valores de entrada precisam estar dentro dos limites utilizados no treinamento. Para utilizar a regressão linear, depois do treinamento, basta aplicar o modelo aprendido. A Equação 1 apresenta o modelo treinado neste experimento pelo algoritmo regressão linear:

$$M = 0.0054A - 0.0242P + 8.0885 \quad (1)$$

onde M é a massa, A é a área e P é o perímetro do alevino.

Os desempenhos das arquiteturas e técnicas de aprendizagem profunda utilizadas na abordagem de classificação são apresentados na Tabela 3. Como destacado, a arquitetura ResNet50, com a técnica de transferência de aprendizagem sem ajuste fino, obteve o melhor resultado. Podemos observar que a técnica de transferência de aprendizagem sem ajuste fino obteve melhores resultados em relação a técnica sem transferência de aprendizagem, evidenciando que, para este problema, é melhor importar os pesos do problema da IMAGENET [39] do que iniciar os pesos com valores aleatórios.

A Figura 9 representa a matriz de confusão da arquitetura ResNet50, com a técnica de transferência

Tabela 3: Resultados do experimento de classificação por aprendizagem profunda.

Arquitetura	Técnica	Acurácia		
		Treino	Validação	Teste
ResNet50	Sem transferência de aprendizagem	0.3694	0.4870	0.3292
	Com transferência de aprendizagem	0.6833	0.6913	0.6708
	Ajuste fino 25%	0.7028	0.4391	0.3917
	Ajuste fino 50%	0.7000	0.4522	0.3917
	Ajuste fino 75%	0.4958	0.3783	0.3333
	Ajuste fino 100%	0.5903	0.4000	0.4625
InceptionV3	Sem transferência de aprendizagem	0.4153	0.5000	0.4583
	Com transferência de aprendizagem	0.5944	0.6000	0.6542
	Ajuste fino 25%	0.3750	0.4174	0.3000
	Ajuste fino 50%	0.5542	0.5304	0.5333
	Ajuste fino 75%	0.4208	0.5304	0.5417
	Ajuste fino 100%	0.4208	0.4826	0.4625
Xception	Sem transferência de aprendizagem	0.5000	0.6043	0.6167
	Com transferência de aprendizagem	0.6486	0.6565	0.6375
	Ajuste fino 25%	0.4903	0.4696	0.3750
	Ajuste fino 50%	0.6125	0.5609	0.5750
	Ajuste fino 75%	0.5556	0.5391	0.5250
	Ajuste fino 100%	0.5375	0.5304	0.4667
VGG16	Sem transferência de aprendizagem	0.3917	0.4478	0.3917
	Com transferência de aprendizagem	0.5306	0.5792	0.5507
	Ajuste fino 25%	0.5681	0.6348	0.6125
	Ajuste fino 50%	0.4889	0.5522	0.5625
	Ajuste fino 75%	0.5014	0.5565	0.5500
	Ajuste fino 100%	0.4097	0.5261	0.4625
VGG19	Sem transferência de aprendizagem	0.3917	0.4478	0.3917
	Com transferência de aprendizagem	0.4972	0.5652	0.5833
	Ajuste fino 25%	0.5014	0.5783	0.5500
	Ajuste fino 50%	0.4889	0.5740	0.5667
	Ajuste fino 75%	0.4833	0.5652	0.5417
	Ajuste fino 100%	0.3486	0.5087	0.4417

de aprendizagem sem a utilização do ajuste fino, respectivamente. Podemos observar, assim como ocorreu com os algoritmos rasos, que os erros de predição se concentram em torno da diagonal principal.

Um dos problemas observados nos experimentos realizados é a flexão do corpo dos alevinos durante a natação. Essa flexão faz com que o mesmo espécime seja observado com formas, tamanhos e orientações muito diferentes, dependendo do quadro extraído do vídeo. Para o experimento utilizando os algoritmos rasos, nos quais são utilizadas a área e perímetro do contorno, um mesmo espécime, com a mesma massa, pode possuir valores bem distintos para essas características. Como os algoritmos recebem os dados de cada quadro como sendo um espécime diferente, não há conexão entre os dados de um mesmo

alevino, dificultando a criação de um modelo de maior eficiência.

Podemos observar também uma diferença de 9% de acurácia da arquitetura de rede neural convolucional que obteve o melhor desempenho sobre o algoritmo J48, algoritmo raso que obteve o melhor resultado, evidenciando que, para este problema, as redes neurais convolucionais são melhores que os algoritmos rasos com as características, área e perímetro, extraídas.

6. Conclusão

A partir dos resultados alcançados, concluímos que na abordagem de classificação com algoritmos rasos o melhor resultado foi obtido pelo algoritmo de classificação J48, o qual atingiu uma acurácia de 58.2%.

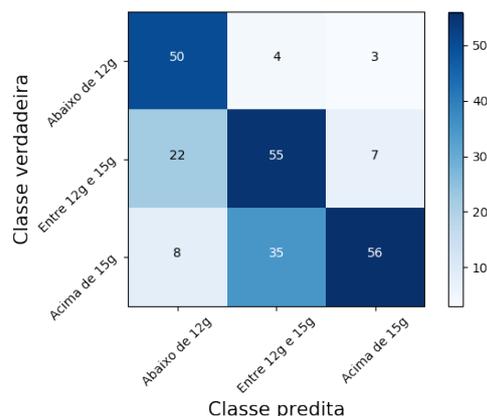


Figura 9: Matriz de confusão da arquitetura ResNet50 com transferência de aprendizagem.

Já na abordagem de regressão com os algoritmos rasos, o algoritmo que obteve o melhor resultado foi o de regressão linear, o qual obteve um coeficiente de correlação de 0.5876 e a média dos erros absolutos de 1.5127 gramas. Das arquiteturas de redes neurais convolucionais testadas, a ResNet50, com a técnica de transferência de aprendizagem sem ajuste fino, foi a que obteve o melhor resultado, alcançando acurácias de 68.33%, 69.13% e 67.08% no treino, validação e teste, respectivamente. Outro resultado importante é a diferença de 9% da rede neural convolucional ResNet50 em relação ao algoritmo J48, algoritmo que obteve o melhor resultado, indicando que as redes neurais convolucionais são melhores se comparadas aos algoritmos rasos utilizando as características, área e perímetro, extraídas.

Agradecimentos

Este trabalho recebe apoio da Universidade Católica Dom Bosco, UCDB, do Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPQ, da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, CAPES, e da Fundação de Apoio ao Desenvolvimento do Ensino, Ciência e Tecnologia do Estado de Mato Grosso do Sul, Fundect, através do termo de outorga 131/2016.

Referências

- [1] FAO, Pma e oms (2017) o estado da segurança alimentar e nutricional no mundo 2017: Construindo resiliência para a paz e segurança alimentar, FAO.
- [2] FAO, Oecd-fao agricultural outlook 2017-2026 (2017).

- [3] M. F. Brabo, L. F. S. Pereira, J. V. M. Santana, D. A. V. Campelo, G. C. Veras, Cenário atual da produção de pescado no mundo, no brasil e no estado do pará: ênfase na aquicultura/current scenario of fish production in the world, brazil and pará state: emphasis on aquaculture, *Acta of Fisheries and Aquatic Resources* 4 (2) (2016) 50–58.
- [4] P. A. Buckup, N. A. Menezes, M. S. Ghazzi, Catálogo das espécies de peixes de água doce do Brasil, Vol. 1, Museu Nacional Rio de Janeiro, 2007.
- [5] D. R. Andrade, G. S. Yasui, Manejo da reprodução natural e artificial e sua importância na produção de peixes no brasil, *Revista Brasileira de Reprodução Animal* 27 (2) (2003) 166–172.
- [6] FAO, Novo relatório da fao aponta que produção da pesca e aquicultura no brasil deve crescer mais de 100% até 2025, <http://www.fao.org/brasil/noticias/detail-events/pt/c/423722/> Acessado em: 01/04/2017 (2016).
- [7] FAO, El estado mundial de la pesca y la acuicultura, LA, CONTRIBUCIÓN A and LA, ALIMENTARIA Y (2016) 224.
- [8] L. Sidonio, I. Cavalcanti, L. Capanema, R. Morch, G. Magalhães, J. Lima, V. Burns, A. J. Alves Júnior, R. Mungiolli, Panorama da aquicultura no brasil: desafios e oportunidades, *BNDES setorial* 35 (2012) 421–463.
- [9] L. Fan, Y. Liu, Automate fry counting using computer vision and multi-class least squares support vector machine, *Aquaculture* 380 (2013) 91–98.
- [10] Y. Duan, L. H. Stien, A. Thorsen, Ø. Karlsen, N. Sandlund, D. Li, Z. Fu, S. Meier, An automatic counting system for transparent pelagic fish eggs based on computer vision, *Aquacultural engineering* 67 (2015) 8–13.
- [11] C. Costa, M. Scardi, V. Vitalini, S. Cataudella, A dual camera system for counting and sizing northern bluefin tuna (*thunnus thynnus*; linnaeus, 1758) stock, during transfer to aquaculture cages, with a semi automatic artificial neural network tool, *Aquaculture* 291 (3-4) (2009) 161–167.
- [12] J. Hernández-Ontiveros, E. Inzunza-González, E. García-Guerrero, O. López-Bonilla, S. Infante-Prieto, J. R. Cárdenas-Valdez, E. Tlelo-Cuautle, Development and implementation of a fish counter by using an embedded system, *Computers and Electronics in Agriculture* 145 (2018) 53–62.
- [13] I. Klapp, O. Arad, L. Rosenfeld, A. Barki, B. Shaked, B. Zion, Ornamental fish counting by non-imaging optical system for real-time applications, *Computers and Electronics in Agriculture* 153 (2018) 126–133.
- [14] M. Saberioon, P. Císař, Automated within tank fish mass estimation using infrared reflection system, *Computers and electronics in agriculture* 150 (2018) 484–492.
- [15] P. Muñoz-Benavent, G. Andreu-García, J. M. Valiente-González, V. Atienza-Vanacloig, V. Puig-Pons, V. Espinosa, Enhanced fish bending model for automatic tuna sizing using computer vision, *Computers and electronics in agriculture* 150 (2018) 52–61.
- [16] V. Atienza-Vanacloig, G. Andreu-García, F. López-García, J. M. Valiente-González, V. Puig-Pons, Vision-based discrimination of tuna individuals in grow-out cages through a fish bending model, *Computers and Electronics in Agriculture* 130 (2016) 142–150.
- [17] V. Puig-Pons, P. Muñoz-Benavent, V. Espinosa, G. Andreu-García, J. M. Valiente-González, V. Estruch, P. Ordóñez, I. Pérez-Arjona, V. Atienza, B. Mèlich, et al., Automatic bluefin tuna (*thunnus thynnus*) biomass estimation during transfers using acoustic and computer vision techniques, *Aquacultural Engineering*.
- [18] L. De Wet, E. Vranken, A. Chedad, J.-M. Aerts, J. Ceunen, D. Berckmans, Computer-assisted image analysis to quantify

- daily growth rates of broiler chickens, *British poultry science* 44 (4) (2003) 524–532.
- [19] G. Pastorelli, M. Musella, M. Zaninelli, F. Tangorra, C. Corino, Static spatial requirements of growing-finishing and heavy pigs, *Livestock Science* 105 (1) (2006) 260–264.
- [20] P. Negretti, G. Bianconi, A. Finzi, Visual image analysis to estimate morphological and weight measurements in rabbits., *World Rabbit Science* 15 (1) (2010) 37–41.
- [21] S. Ozkaya, Y. Bozkurt, The relationship of parameters of body measures and body weight by using digital image analysis in pre-slaughter cattle, *Archiv fur Tierzucht* 51 (2) (2008) 120.
- [22] F. Odone, E. Trucco, A. Verri, A trainable system for grading fish from images, *Applied Artificial Intelligence* 15 (8) (2001) 735–745.
- [23] J. Lines, R. Tillett, L. Ross, D. Chan, S. Hockaday, N. McFarlane, An automatic image-based system for estimating the mass of free-swimming fish, *Computers and Electronics in Agriculture* 31 (2) (2001) 151–168.
- [24] B. Zion, The use of computer vision technologies in aquaculture—a review, *Computers and electronics in agriculture* 88 (2012) 125–132.
- [25] S. Viazzi, S. Van Hoestenbergh, B. Goddeeris, D. Berckmans, Automatic mass estimation of jade perch scortum barcoo by computer vision, *Aquacultural Engineering* 64 (2015) 42–48.
- [26] V. Puig-Pons, V. D. Estruch, V. Espinosa, F. de la Gándara, B. Melich, J. L. Cort, Relationship between weight and linear dimensions of bluefin tuna (*thunnus thynnus*) following fattening on western mediterranean farms, *PloS one* 13 (7) (2018) e0200406.
- [27] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, G. Nadarajan, Automatic fish classification for underwater species behavior understanding, in: *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, ACM, 2010, pp. 45–50.
- [28] B. Zion, V. Alchanatis, V. Ostrovsky, A. Barki, I. Karplus, Real-time underwater sorting of edible fish species, *Computers and Electronics in Agriculture* 56 (1) (2007) 34–45.
- [29] S. J. Prince, *Computer vision: models, learning, and inference*, Cambridge University Press, 2012.
- [30] N. M. Nasrabadi, *Pattern recognition and machine learning*, *Journal of electronic imaging* 16 (4) (2007) 049901.
- [31] Y. Mao, A. Monahan, Linear and nonlinear regression prediction of surface wind components, *Climate Dynamics* (2018) 1–19.
- [32] H. Pandorfi, G. L. P. Almeida, C. Guiselini, *Zootecnia de precisão: princípios básicos e atualidades na suinocultura*, *Revista Brasileira de Saúde e Produção Animal* 13 (2).
- [33] S. Ozkaya, The prediction of live weight from body measurements on female holstein calves by digital image analysis, *The Journal of Agricultural Science* 151 (4) (2013) 570–576.
- [34] S. Tasdemir, A. Urkmez, S. Inal, Determination of body measurements on the holstein cows using digital image analysis and estimation of live weight with regression analysis, *Computers and electronics in agriculture* 76 (2) (2011) 189–197.
- [35] X. Song, E. Bokkers, P. van der Tol, P. G. Koerkamp, S. van Mourik, Automated body weight prediction of dairy cows using 3-dimensional vision, *Journal of dairy science* 101 (5) (2018) 4448–4459.
- [36] C. Shi, G. Teng, Z. Li, An approach of pig weight estimation using binocular stereo system based on labview, *Computers and Electronics in Agriculture* 129 (2016) 37–43.
- [37] A. L. Zhang, B. P. Wu, C. T. Wuyun, D. X. Jiang, E. C. Xuan, F. Y. Ma, Algorithm of sheep body dimension measurement and its applications based on image analysis, *Computers and Electronics in Agriculture* 153 (2018) 33–45.
- [38] R. Kohavi, et al., A study of cross-validation and bootstrap for accuracy estimation and model selection, in: *Ijcai*, Montreal, Canada, 1995, pp. 1137–1145.
- [39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, *IEEE*.

Considerações Finais

Nesta seção serão apresentados os principais resultados obtidos pela pesquisa realizada e a sugestão de ajustes nos experimentos e de trabalhos futuros.

6.1 Principais Resultados

Foram realizados dois experimentos, um de contagem, utilizando o *software* do contador de alevinos, e um de estimativa da massa, em duas abordagens, classificação e regressão, utilizando algoritmos rasos. O experimento de estimativa da massa se estendeu, na abordagem de classificação, em testes de arquiteturas de redes neurais convolucionais, sendo que os resultados dos mesmos foram comparados aos dos algoritmos rasos. Para esta finalidade, foram criados dois bancos de imagens, o FISCHV155V e o FISHCV2016FSEG, para os experimentos de contagem e estimativa da massa, respectivamente. Os experimentos foram descritos em dois artigos, os quais serão submetidos, depois de revisados.

No experimento de contagem, foram testados 324 combinações de parâmetros para o *software* do contador de alevinos. A partir dos resultados alcançados, concluímos que a combinação de parâmetros, área do *blob*, limiar do *blob*, distância entre *blobs*, limiar candidato, desvio padrão e área mínima, com os valores 62, 33, 70, 1 1,1 e 1,7, respectivamente, apresentou a menor média do erros ab-

solutos, 5,99, e a menor raiz da média dos erros ao quadrado, 9,96, perfazendo assim, a melhor combinação de parâmetros dentro das combinações testadas. Outro resultado importante observado é que, para os valores escolhidos, a variação de alguns parâmetros não alteram as métricas gerais.

O experimento de estimativa da massa foi dividido em duas partes. Na primeira parte foram realizados testes de classificação e regressão através de algoritmos rasos. Para isso, foram extraídas características, área e perímetro, dos contornos dos alevinos. Na segunda parte foram testadas arquiteturas de redes convolucionais, em conjunto com transferência de aprendizagem e ajuste fino, na classificação dos alevinos. A partir dos resultados alcançados, concluímos que na abordagem de classificação com algoritmos rasos o melhor resultado foi obtido pelo algoritmo J48, o qual atingiu uma acurácia de 58.2%. Já na abordagem de regressão com os algoritmos rasos, o algoritmo que obteve o melhor resultado foi o de regressão linear, o qual obteve um coeficiente de correlação de 0.5876 e a média dos erros absolutos de 1.5127 gramas. Das arquiteturas de redes neurais convolucionais testadas, a ResNet50, com a técnica de transferência de aprendizagem sem ajuste fino, foi a que obteve o melhor resultado, alcançando acurácias de 68.33%, 69.13% e 67.08% no treino, validação e teste, respectivamente. Outro resultado importante é a diferença de 9% da rede neural convolucional ResNet50 em relação ao algoritmo J48, algoritmo que obteve o melhor resultado, indicando que as redes neurais convolucionais são melhores se comparadas aos algoritmos rasos utilizando as características, área e perímetro, extraídas.

6.2 *Trabalhos Futuros*

Um dos problemas observados no experimento de estimativa da massa realizado é a flexão do corpo dos alevinos durante a natação. Essa flexão faz com que o mesmo espécime seja observado com formas, tamanhos e orientações muito diferentes, dependendo do quadro extraído do vídeo. Para o experimento utilizando os algoritmos rasos, nos quais são utilizadas a área e perímetro do contorno, um mesmo espécime, com a mesma massa, pode possuir valores bem distintos para essas características. Como os algoritmos recebem os dados de cada quadro como sendo um espécime diferente, não há conexão entre os dados de um mesmo alevino, dificultando a criação de um modelo de maior eficiência. Uma solução

proposta seria a mudança na metodologia do experimento, fazendo a média das áreas e perímetros de cada espécime ou pegando os maiores valores.

Na construção dos novos bancos de imagens, explorar ajustes de foco, pois os bancos criados neste trabalho foram feitos com foco automático, resultando em problemas de desfocagem em alguns vídeos. Outro ponto a se observar na construção de novos bancos seria a captura de todas as faixas possíveis de tamanhos, para que modelos mais robustos de classificação e regressão possam ser criados.

A realização da regressão com redes neurais convolucionais seria bem interessante para comparar com os resultados da regressão com algoritmos rasos, da mesma maneira como foi feito com a classificação dos alevinos pela massa.

Referências Bibliográficas

- Abdel-Hamid, O., Deng, L., and Yu, D. (2013). Exploring convolutional neural network structures and optimization techniques for speech recognition. In *Interspeech*, volume 2013, pages 1173–5. 22
- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282. 11, 12
- Aha, D. W., Kibler, D., and Albert, M. K. (1991). Instance-based learning algorithms. *Machine learning*, 6(1):37–66. 19
- Ahmed, F., Al-Mamun, H. A., Bari, A. H., Hossain, E., and Kwan, P. (2012). Classification of crops and weeds from digital images: A support vector machine approach. *Crop Protection*, 40:98–104. 17
- Andrade, D. R. and Yasui, G. S. (2003). Manejo da reprodução natural e artificial e sua importância na produção de peixes no brasil. *Revista Brasileira de Reprodução Animal*, 27(2):166–172. 1
- Arel, I., Rose, D. C., and Karnowski, T. P. (2010). Deep machine learning—a new frontier in artificial intelligence research [research frontier]. *IEEE computational intelligence magazine*, 5(4):13–18. 21
- Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM. 16

- Brabo, M. F., Pereira, L. F. S., Santana, J. V. M., Campelo, D. A. V., and Veras, G. C. (2016). Cenário atual da produção de pescado no mundo, no brasil e no estado do pará: ênfase na aquicultura/current scenario of fish production in the world, brazil and pará state: emphasis on aquaculture. *Acta of Fisheries and Aquatic Resources*, 4(2):50–58. 1
- Buckup, P. A., Menezes, N. A., and Ghazzi, M. S. (2007). *Catálogo das espécies de peixes de água doce do Brasil*, volume 1. Museu Nacional Rio de Janeiro. 1
- Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27. 19
- Deng, L., Yu, D., et al. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387. 20, 21
- FAO (2016a). El estado mundial de la pesca y la acuicultura. *LA, CONTRIBUCIÓN A and LA, ALIMENTARIA Y*, page 224. 2
- FAO (2016b). Novo relatório da fao aponta que produção da pesca e aquicultura no brasil deve crescer mais de 100% até 2025. <http://www.fao.org/brasil/noticias/detail-events/pt/c/423722/>
Acessado em: 01/04/2017. 1
- FAO (2017). *Oecd-fao agricultural outlook 2017-2026*. 1
- Fukushima, K. and Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer. 21
- Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422. 16
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18. 15, 16
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108. 12

- He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385. 24
- Hu, Y., Chang, H., Nian, F., Wang, Y., and Li, T. (2016). Dense crowd counting from still images with convolutional neural networks. *Journal of Visual Communication and Image Representation*, 38:530–539. 24
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105. 24
- Lab, L. (2015). Deep learning tutorial. *University of Montreal*. 20
- LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., and Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404. 21, 22, 24
- Mao, Y. and Monahan, A. (2018). Linear and nonlinear regression prediction of surface wind components. *Climate Dynamics*, pages 1–19. 3
- Michie, D., Nuñez, M., Podgorelec, V., Kokol, P., Stiglic, B., and Rozman, I. (1993). Quinlan, jr c4. 5: Programs for machine learning. 19
- Nasrabadi, N. M. (2007). Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4):049901. 2
- Neter, J., Kutner, M. H., Nachtsheim, C. J., and Wasserman, W. (1996). *Applied linear statistical models*, volume 4. Irwin Chicago. 20
- Parker, J. R. (2010). *Algorithms for image processing and computer vision*. John Wiley & Sons. 21
- Piccardi, M. (2004). Background subtraction techniques: a review. In *Systems, man and cybernetics, 2004 IEEE international conference on*, volume 4, pages 3099–3104. IEEE. 13
- Platt, J. (1998). Sequential minimal optimization: A fast algorithm for training support vector machines. *Microsoft*. 17

- Prince, S. J. (2012). *Computer vision: models, learning, and inference*. Cambridge University Press. 2, 9, 10, 21
- Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier. 18
- Ristic, B., Arulampalam, S., and Gordon, N. (2003). *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house. 13
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252. 21, 24
- Sahoo, P. K., Soltani, S., and Wong, A. K. (1988). A survey of thresholding techniques. *Computer vision, graphics, and image processing*, 41(2):233–260. 11
- Shapiro, L. and Stockman, G. (2001). *Computer Vision*. October, 2004(October):608. 9, 21
- Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., and Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298. 22
- Sidonio, L., Cavalcanti, I., Capanema, L., Morch, R., Magalhães, G., Lima, J., Burns, V., Alves Júnior, A. J., and Mungioli, R. (2012). Panorama da aquicultura no brasil: desafios e oportunidades. *BNDES setorial*, 35:421–463. 2
- Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. 24
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., et al. (2015a). Going deeper with convolutions. In *CVPR 2015*. Cvpr. 24
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2015b). Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567. 24

- Szeliski, R. (2010). Computer Vision : Algorithms and Applications. *Computer*, 5:832. 10
- Thrun, S. and Norvig, P. (2011). Introduction to artificial intelligence. *Retrieved November*, 5:2011. 15
- Vapnik, V. N. and Vapnik, V. (1998). *Statistical learning theory*, volume 1. Wiley New York. 16
- Yuan, J., Ni, B., and Kassim, A. A. (2014). Half-cnn: a general framework for whole-image regression. *arXiv preprint arXiv:1412.6885*. 24
- Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer. 24
- Zion, B. (2012). The use of computer vision technologies in aquaculture—a review. *Computers and electronics in agriculture*, 88:125–132. 2