

RECONHECIMENTO DE PEIXES USANDO REDES NEURAS CONVOLUCIONAIS PARA PREVENÇÃO DE FRAUDE

FISH RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS FOR FRAUD PREVENTION

Rhayane da Silva Monteiro*

Mário Wedney de Lima Moreira**

Glácio Souza Araújo***

RESUMO

A fraude, a identificação incorreta e a adulteração de alimentos, involuntárias ou propositadas, são uma preocupação mundial e crescente. A aquicultura e a pesca são reconhecidas como um dos setores mais vulneráveis à fraude alimentar. Além disso, uma série de riscos relacionados à saúde e à desconfiança entre consumidor e mercado exigem que esse setor desenvolva uma solução eficaz para o controle dessa prática ilícita. A identificação de espécies representa um aspecto essencial para expor falsificações comerciais. As *convolutional neural networks* (CNNs) são uma das ferramentas mais poderosas para tarefas de reconhecimento e classificação de imagens. Dessa forma, o objetivo deste estudo é propor um modelo de reconhecimento de espécies de peixes baseado em CNNs. Os resultados obtidos mostram um algoritmo com acurácia de 86%, proporcionando uma solução eficaz para o combate à fraude em pescado.

Palavras-chave: Fraude alimentar. Reconhecimento de peixes. Redes neurais convolucionais.

ABSTRACT

Fraud, misidentification, and adulteration of food, whether unintentional or purposeful, are a worldwide and growing concern. Aquaculture and fisheries are recognized as one of the sectors most vulnerable to food fraud. Besides, a series of risks related to health and distrust between consumer and popular market that this sector develop an effective solution for fraud control. Species identification is an essential aspect to expose commercial fraud. Convolutional neural networks (CNNs) are one of the most powerful tools for image recognition and classification

* Graduada em Bacharelado em Ciência da Computação, Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Aracati, Ceará, Brasil. *E-mail:* rhayane.monteiro@ifce.edu.br

** Doutor em Engenharia Informática, Docente do Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Aracati, Ceará, Brasil. *E-mail:* mario.wedney@ifce.edu.br

*** Doutor em Engenharia de Pesca, Docente do Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Aracati, Ceará, Brasil. *E-mail:* glacio@ifce.edu.br

tasks. Thus, the objective of this study is to propose a model of recognition of fish species based on CNNs. The results obtained show an algorithm with an accuracy of 86%, providing an effective solution to prevent fish fraud.

Keywords: Food fraud. Fish Recognition. Convolutional Neural Networks.

1 INTRODUÇÃO

A substituição, de forma deliberada e intencional, a rotulagem incorreta, a adulteração ou falsificação de alimentos, matérias-primas, ingredientes ou embalagens se define como fraude de produto, conforme o guia da *International Featured Standards* (IFS) ¹. A fraude em alimentos, por sua vez, é uma crescente preocupação mundial (BÖHME et al., 2019). Com a globalização e sistemas complexos de distribuição, a fraude alimentar tornou-se comum, acarretando impacto de longo alcance e consequências adversas no bem-estar da população. Segundo o Manual de Integridade dos Alimentos, elaborado pela *Food Integrity* ², a adulteração se apresenta como a mais corriqueira, considerando os tipos de fraude em alimentos. Contudo, a autenticação de alimentos abrange muitos outros tópicos, como origem geográfica, identificação de variedades, método de produção, além da detecção de organismos geneticamente modificados. Segundo Song *et. al* (2020, p. 1), as fraudes alimentares são problemas que acarretam impacto significativo em indivíduos, empresas e sociedade, podendo, portanto, causar risco à saúde pública, perda de negócios e redução da confiança do consumidor nas marcas.

Peixes e demais frutos do mar são um dos produtos mais comercializadas no mundo (BÖHME et al., 2019). Conforme Probst (2020, p. 1), em razão de sua atuação em escala global, o setor pesqueiro é frequentemente desafiado pela falta de confiança em várias etapas do processo produtivo e da cadeia de suprimentos. O relatório da *Food and Agriculture Organization of the United Nations* (FAO) afirma que a pesca e a aquicultura são reconhecidas como um dos setores mais vulneráveis à fraude alimentar ³. A adulteração em peixes é cometida quando o peixe é propositadamente colocado no mercado, para ganho financeiro, com a intenção de enganar o cliente. Existem tipos diferentes de fraude que podem ocorrer em vários pontos ao longo da cadeia de fornecimento de peixes. O tipo mais comum envolve a rotulagem intencional e a substituição de espécies. Em contrapartida, a capacidade de verificar a autenticidade e proveniência dos frutos do mar, da captura ao consumo, tornou-se um requisito essencial para práticas comerciais sustentáveis, seguras e compatíveis (VO et al., 2020). Assim, são necessárias ferramentas rápidas e fáceis para a avaliação da autenticidade desses produtos para enfrentar as fraudes econômicas ao longo de toda a cadeia de fornecimento.

A identificação de espécies de peixes representa um aspecto fundamental para expor fraudes comerciais (ROSSI et al., 2016). Anteriormente, as espécies de peixes eram identificadas

¹ Cf. <https://ifs-productintegrity.com/wp-content/uploads/2020/09/IFS_Guideline_Product_Fraud_Mitigation_V2_EN.pdf>.

² Disponível em: <<https://secure.fera.defra.gov.uk/foodintegrity/index.cfm?sectionid=83>>.

³ Disponível em: <<http://www.fao.org/3/i8791en/i8791EN.pdf>>.

manualmente pela observação e, portanto, era necessário lembrar ou estudar várias características dos peixes para reconhecê-los. Além disso, a identificação de espécies de peixes usando apenas a aparência é difícil para os seres humanos, porque espécies de uma mesma família apresentam aparência semelhante, além do ambiente e as condições ao redor, como os *habitats*, podem mudar a cor dos peixes (MASUDA; JUKEI; HASEGAWA, 2020). Nesse contexto, percebe-se a necessidade de técnicas capazes de realizar uma identificação correta, automática e rápida de peixes. Ao analisar diferentes sistemas de reconhecimento de espécies para evitar a falsificação, vários critérios de decisão devem ser considerados. Isso inclui tempo de resposta, precisão, resolução e recursos. Conforme Hridayami *et. al* (2019, p. 124), a tecnologia avançada, aliada com a inteligência artificial e métodos de aprendizado profundo (*deep learning*, em inglês), facilitam a tarefa de reconhecimento de espécies de peixes.

Deep learning é definida como um método de aprendizado por representação, de maneira a preservar a informação de entrada dos dados, porém transformando-a de forma que se torne útil para utilização. Assim, as técnicas de *deep learning* refinam a representação em vários níveis, utilizando uma rede neural artificial profunda e composta por múltiplas camadas de neurônios. Devido à eficiente capacidade de aprendizado de características, muitos problemas complexos podem ser resolvidos de maneira rápida e eficaz (ZHOU *et al.*, 2019). A *deep learning* possui várias aplicações, como reconhecimento automático de fala, processamento de linguagem natural e reconhecimento de imagens (RACHEL; VARALAKSHMI, 2019). As redes neurais convolucionais (*convolutional neural networks* (CNNs), em inglês) são um dos métodos mais populares de *deep learning* (HRIDAYAMI; PUTRA; WIBAWA, 2019). Segundo Sanghvi *et al.* (2020, p. 8), este método continua sendo a escolha mais procurada pelos cientistas da computação para reconhecimento, processamento e classificação de imagens.

As CNNs são algoritmos de *deep learning* que recebem uma imagem como entrada, atribuem importância ou pesos a vários aspectos dela e é capaz de diferenciar uma da outra. O nome desse tipo de rede se deve à operação de convolução. A convolução consiste na extração de características úteis dos dados de entrada, nesse caso, das imagens. Além da convolução, uma CNN é composta pelas camadas de entrada, camada de agrupamento (*pooling*, em inglês), camada totalmente conectada e camada de saída. As CNNs são normalmente usadas para classificação e reconhecimento de imagens devido à sua alta precisão (SANGHVI *et al.*, 2020). Além disso, elas são flexíveis e podem se adaptar aos novos dados recebidos à medida que o conjunto de dados cresce. Rachel and Varalakshmi (2019, p. 48) afirmam que a principal vantagem das CNNs é que elas detectam automaticamente as características sem nenhuma intervenção humana.

Neste contexto, este trabalho propõe um modelo de reconhecimento de peixes para prevenção de fraude baseado em CNNs. Em seguida, o *dataset* construído neste estudo foi submetido às arquiteturas de CNN para a classificação. Posteriormente, foram realizados testes e experimentos para avaliar e validar o modelo com melhor desempenho. Finalmente, um protótipo de uma aplicação *web* é apresentado a partir do modelo de CNN. Os resultados mostraram um algoritmo com acurácia de 86%, evidenciando a relevância da aplicação para a prevenção de fraude em peixes.

O trabalho organiza-se em seções, apresentando a seguinte sequência, a saber, a seção 2 apresenta os conceitos importantes para o entendimento da proposta; a seção 3 expõe os trabalhos que relacionam-se à temática deste artigo. Já os materiais e métodos são mostrados na seção 4 e a seção 5 é utilizada para resultados e discussões. O artigo finaliza na seção 6 com a conclusão e a proposição de trabalhos futuros.

2 REVISÃO DA LITERATURA

Esta seção apresenta os conceitos necessários para o entendimento da proposta deste trabalho.

2.1 Deep Learning

Deep learning se tornou um tópico de pesquisa em rápida expansão e que foi introduzido com o objetivo de aproximar o aprendizado de máquina da inteligência artificial (WANG et al., 2017). Contudo, a *deep learning* perpassa o aprendizado de máquina clássico adicionando mais complexidade ao modelo. Além disso, ela transforma os dados usando várias funções que permitem a representação de dados de maneira hierárquica, por meio de vários níveis de abstração.

A *deep learning* pode ser entendida como o uso de redes neurais profundas com várias camadas não lineares, nas quais as características são aprendidas com os dados por meio de um procedimento de aprendizado de uso geral. Uma forte vantagem de abordagens baseadas em *deep learning* é o aprendizado de características, *i.e.*, a extração automática de características a partir de dados brutos (KAMILARIS; PRENAFETA-BOLDÚ, 2018). Além disso, ela pode resolver problemas mais complexos de forma rápida, devido aos modelos mais complexos utilizados. Tais modelos podem aumentar a precisão da classificação ou reduzir o erro nos problemas de regressão, desde que existam conjuntos de dados adequadamente grandes descrevendo o problema. Esses fatores possibilitam às estruturas de *deep learning* serem flexíveis e adaptáveis a uma ampla variedade de desafios altamente complexos. Embora métodos baseados nesta abordagem tenham sido usados em muitos campos de pesquisa, o campo mais eficaz é o reconhecimento de objetos (QIN et al., 2016).

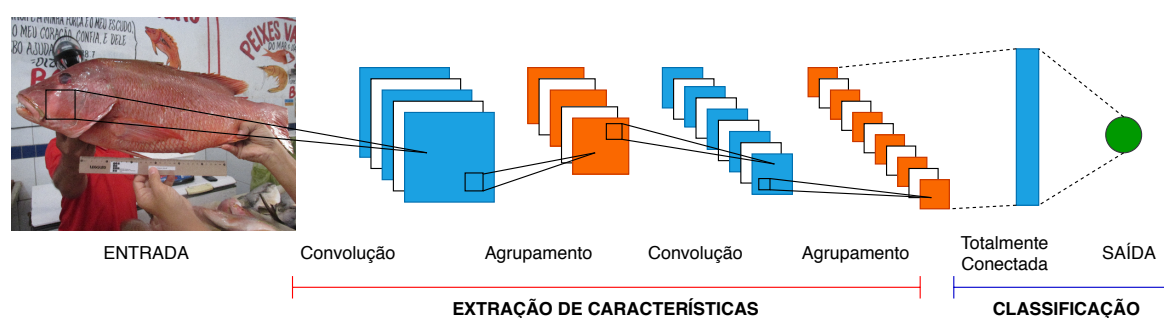
Dentre as diversas abordagens do reconhecimento de objetos, o *deep learning* destaca-se no processamento de imagens e análise de dados, com resultados promissores e grande potencial. Uma das vantagens mais importantes do uso de *deep learning* no processamento de imagens é a necessidade reduzida da engenharia de características, uma vez que estas são localizadas por meio de treinamento e de forma automática. Uma desvantagem desta abordagem é o tempo de treinamento que, geralmente, é mais longo. No entanto, o tempo de teste geralmente é mais rápido do que outros métodos baseados em aprendizado de máquina. A *deep learning* possui diversos tipos de arquitetura, tais como redes neurais recorrentes, redes neurais recursivas e CNNs. A CNN é uma das ferramentas mais poderosas de *deep learning* para a classificação de imagens com base em características visuais (RAUF et al., 2019).

2.2 Convolutional Neural Networks

As CNNs são uma classe de algoritmos de aprendizado profundo usados para analisar dados e um dos melhores algoritmos de aprendizado para a compreensão do conteúdo da imagem. Além disso, elas mostraram desempenho exemplar nas tarefas relacionadas à segmentação, classificação, detecção e recuperação de imagens (KHAN et al., 2020). Algumas das áreas de aplicação das CNNs incluem detecção de objetos, processamento de vídeo, classificação e segmentação de imagens. A poderosa capacidade de aprendizado da CNN profunda deve-se principalmente ao uso de vários estágios de extração de características que podem aprender automaticamente representações dos dados. Além disso, as CNNs necessitam de um tempo menor de pré-processamento comparado a outros algoritmos de classificação, uma vez que ela consegue aprender as características de forma automática, enquanto que nos métodos tradicionais, o processo é feito de forma manual.

Uma CNN consiste em neurônios que carregam pesos, viés e funções de ativação. Semelhante a outras redes neurais, a CNN compreende camadas de entrada, ocultas e de saída, que realizam operações que alteram os dados para investigar as características específicas. Uma arquitetura básica de CNN possui as camadas de entrada, convolução e *pooling*, que alternam-se, camada totalmente conectada e saída, conforme mostra a Figura 1. A CNN recebe uma imagem como entrada. A camada de convolução atua como um extrator de características das imagens de entrada, cuja dimensionalidade é então reduzida pelas camadas de agrupamento. As camadas convolucionais codificam várias características de nível inferior para características mais discriminativas, que transformam uma imagem de entrada em outra, destacando padrões específicos. A camada totalmente conectada atua como classificador, que explora as características de alto nível aprendidas pela CNN para classificar imagens de entrada em classes predefinidas. Em resumo, essa camada recebe um vetor como entrada e produz outro vetor como saída.

Figura 1 – Arquitetura da *convolutional neural network*



Fonte: Elaborada pelos autores (2020).

A disponibilidade de uma grande quantidade de dados e a melhoria da tecnologia de *hardware* acelerou a pesquisa em CNNs, portanto, várias novas arquiteturas foram propostas. Os modelos elencados de CNNs utilizados neste trabalho são abordados no próximo tópico. O critério de escolha destes modelos baseia-se nas redes mais recentes do estado da arte na área. Além disso, essas arquiteturas são facilmente encontradas na *Internet*, o que facilita o trabalho

ao utilizá-las.

2.2.1 VGG16

Simonyan e Zisserman (2014, p. 5) propuseram a arquitetura VGG, que apresentava um princípio de *design* simples e eficaz para as arquiteturas de CNN. O VGG16 possui 23 camadas para simular a relação de profundidade com a capacidade representacional da rede. Esta CNN mostrou bons resultados tanto para classificação de imagens quanto para problemas de localização, porém, é comumente reconhecida devido à sua simplicidade, topologia homogênea e maior profundidade (KHAN et al., 2020).

2.2.2 InceptionV3

A ideia do InceptionV3 era reduzir o custo computacional de redes profundas sem afetar a generalização. Para tanto, Szegedy *et al.* (2016, p. 4) substituíram filtros de tamanho grande por filtros pequenos e assimétricos e usou a convolução 1×1 como um gargalo antes dos filtros grandes. Tais características tornaram-se as principais contribuições desta arquitetura. Esta CNN apresenta 159 camadas de profundidade.

2.2.3 DenseNet121

Elaborada por Huang *et al.*, a DenseNet foi proposta para resolver o problema do gradiente de desaparecimento (HUANG et al., 2017). Para isso, esta CNN usou conectividade de camada cruzada, a qual conectou cada camada anterior à próxima camada de chegada de forma *feed-forward*. Assim, mapas de características de todas as camadas anteriores foram usados como entradas em todas as camadas subsequentes. Como o DenseNet concatena as características da camada anterior em vez de adicioná-las, a rede pode ganhar a capacidade de diferenciar explicitamente entre as informações que são adicionadas à rede e as que são preservadas (KHAN et al., 2020). Esta arquitetura apresenta 121 camadas de profundidade.

2.2.4 MobileNet

A MobileNet foi projetada por Howard *et al.* (2017) para ser utilizada em dispositivos móveis. Para isso, esta CNN usa convoluções separáveis em profundidade (HOWARD et al., 2017). Tal fato reduz significativamente o número de parâmetros quando comparado a redes com convoluções regulares com a mesma profundidade. Isso resulta em redes neurais profundas leves. Ao definir a rede em termos tão simples, é possível explorar facilmente a estrutura dela e delinear uma boa rede (KHASOGGI; ERMATITA; SAHMIN, 2019).

2.2.5 Xception

O Xception foi proposto por Chollet (2017) e pode ser considerado uma arquitetura de início extrema, que explora a ideia de convolução separável em profundidade (CHOLLET, 2017).

Esta arquitetura modificou o bloco inicial original tornando-o mais amplo e substituindo as diferentes dimensões espaciais por uma única dimensão, seguida por uma convolução 1×1 para regular a complexidade computacional. O Xception possui 126 camadas de profundidade. Embora a estratégia de transformação adotada por esta CNN não reduza o número de parâmetros, torna o aprendizado mais eficiente e resulta em melhor desempenho (KHAN et al., 2020).

3 TRABALHOS RELACIONADOS

Diversas abordagens usando CNNs foram propostas para o reconhecimento de espécies de peixes e que alcançaram resultados promissores. Nesta seção, serão apresentados alguns trabalhos relacionados à temática deste artigo.

Em seu trabalho, Santos and Gonçalves (2019) apresentaram um método baseado em CNNs para o reconhecimento de espécies de peixes do Pantanal (SANTOS; GONÇALVES, 2019). A CNN, composta por três ramos que classificam os peixes em família e ordem, é proposta com o objetivo de melhorar o reconhecimento de espécies com características semelhantes. Os resultados no conjunto de dados mostraram que o método proposto obteve uma precisão de 0.873 e 0.864 utilizando uma CNN tradicional. Os autores realizaram o reconhecimento de 68 espécies de peixes. Apesar de apresentar uma variabilidade de imagens no *dataset*, estimulando a CNN a se adaptar a diferentes entradas de dados, o foco em extrair imagens em ambientes reais foi desviado. Assim, o modelo pode obter desempenho inferior ao receber imagens de baixa qualidade, em ambientes diferentes das imagens apresentadas, entre outras situações.

Masuda *et al.* (2020) propuseram um método avançado de identificação de espécies de peixes que utiliza imagens de peixes e suas características morfológicas. Os autores desenvolveram um modelo de múltiplas entradas que apresenta imagens e, adicionalmente, dados de características, visando tornar-se um sistema avançado de identificação de espécies de peixes. Para mostrar a precisão do modelo proposto, o trabalho comparou com modelos que usavam apenas imagens ou apenas caracteres morfológicos. Como resultado, foi esclarecido que o uso de dois ou mais tipos de caracteres morfológicos proporcionava maior precisão do que o uso apenas de imagens (MASUDA; JUKEI; HASEGAWA, 2020). Os autores trabalharam com a identificação de nove espécies de peixes. Contudo, conforme exposto no trabalho, o número de imagens por espécie é desequilibrado, de forma que algumas espécies tem cerca de o dobro ou o triplo de imagens de outras espécies, ocasionando *overfitting*, de maneira que o algoritmo classifica erroneamente determinada espécie de peixe por apresentar poucos dados de treinamento.

O trabalho de Rauf *et al.* (2019) mostrou uma estrutura de *deep learning* baseada no método CNN para identificação de espécies de peixes. A arquitetura de CNN proposta contém 32 camadas profundas para extrair características valiosas e discriminatórias da imagem. Para testar o desempenho do modelo, os autores elaboraram um conjunto de dados denominado Fish-Pak, contendo 915 imagens. Com o objetivo de validar a CNN proposta, os autores realizaram comparações com outras estruturas de *deep learning*, envolvendo o VGG16, LeNet5, AlexNet, GoogleNet, e ResNet50. Análises empíricas revelaram que o método proposto alcança desempe-

nho satisfatório e supera os métodos existentes (RAUF et al., 2019). No *dataset* desta pesquisa, seis espécies de peixes foram consideradas, segmentando as imagens em regiões específicas que caracterizam cada espécie estudada. Contudo, o *dataset* também apresenta-se desequilibrado. Isso pode acarretar em um algoritmo inflexível, tornando-o ineficaz para determinadas espécies de peixes.

Vo *et al.* (2020) propõe a utilização de uma CNN em conjunto com técnicas de processamento de imagem para classificar automaticamente lagostas, objetivando o controle de fraude durante a cadeia de comercialização (VO et al., 2020). Ao usar uma CNN, a pesquisa visa aprimorar as pesquisas anteriores sobre classificação de lagostas, estabelecendo um método confiável e flexível de rastreabilidade para atender a diferentes contextos da cadeia de suprimentos. Embora o foco da pesquisa seja o reconhecimento de lagostas, o trabalho gera importantes contribuições. Na abordagem proposta, um modelo Mask-RCNN pré-treinado foi adotado para extrair regiões de interesse das imagens de lagosta. Para provar a eficácia, o método proposto foi validado em um conjunto de dados de imagem coletado em um processador de lagosta e testado no ambiente de aplicativos móveis. Como desafio, evidencia-se que o *dataset* é reduzido, sendo o conjunto de treinamento composto por 150 imagens e o conjunto de validação por 90 imagens. Além disso, a captura das imagens foi realizada por um conjunto de três câmeras colocadas de cima para baixo e também definiu-se a distância adequada para tirar a foto e, assim, o algoritmo ser capaz de reconhecer a lagosta.

Rossi *et al.* (2016) apresentaram o FishAPP, uma infraestrutura baseada em nuvem para reconhecimento de espécies de peixes. A abordagem se dá por um aplicativo móvel desenvolvido para Android e iOS, permitindo ao usuário tirar fotos de um peixe inteiro e enviá-las para análise remota. Um sistema em nuvem realiza o processamento de imagens e, em seguida, é submetido a uma CNN capaz de analisar as imagens obtidas e executar a classificação em espécies de peixes. Os resultados preliminares obtidos a partir do conjunto de dados disponíveis forneceram resultados encorajadores (ROSSI et al., 2016). O trabalho impõe, ao usuário, uma série de diretrizes ao tirar a foto do peixe, tais como a foto deve incluir o peixe completo, este deve ser fotografado de lado e a barbatana caudal deve ser posta de maneira anatômica e relaxada. Tais fatores restringem a relevância da aplicação, por não se adequar a diferentes e possíveis cenários da vida real.

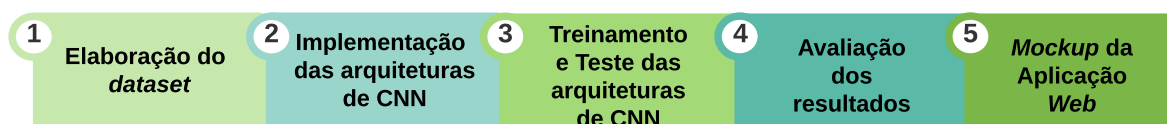
A partir dos trabalhos apresentados, percebe-se ainda desafios para a tarefa de reconhecimento de peixes. São eles *i)* *dataset* com número de imagens limitadas; *ii)* desequilíbrio no quantitativo de imagens por espécie e *iii)* restrição de imagens em cenários reais e sob condições restritas. Diante disso, o objetivo deste estudo é propor um modelo para combate à fraude em pescado utilizando CNNs, contribuindo assim na superação dos desafios elencados.

4 MATERIAIS E MÉTODOS

Nesta seção, são apresentados os materiais e métodos necessários para a implementação da proposta deste trabalho.

A Figura 2 mostra as etapas realizadas para alcançar o objetivo desta pesquisa. Tais fases compreendem cinco atividades principais, a saber, elaboração do *dataset* (1), implementação das arquiteturas de CNN (2), treinamento e teste das arquiteturas (3), avaliação dos resultados (4) e *mockup* da aplicação *web* (5). Essas etapas são explicadas detalhadamente a seguir.

Figura 2 – Fluxograma das etapas do trabalho



Fonte: Elaborada pelos autores (2020).

A etapa 01 compreende a elaboração do *dataset*. Para isso, construiu-se um conjunto de dados com 338 imagens, divididas em 08 espécies de peixes marinhos, comumente encontrados no Litoral Leste do Ceará. As imagens foram capturadas entre agosto de 2019 e agosto de 2020 por meio dos dispositivos *Câmera Canon*, *Smartphones Iphone 7 Plus* e *Xiaomi Redmi Note 8*. Para além disto, as fotos foram coletadas em estabelecimentos de venda de pescados, na cidade de Aracati, CE. A Figura 3 mostra alguns exemplos de imagens no conjunto de dados. Estas imagens geralmente incluem outros elementos além dos peixes, tais como objetos, corpo do ser humano, entre outros. Essas variações incentivam o modelo a selecionar características de identificação que não dependem da orientação ou o foco da imagem que está sendo classificada.

Figura 3 – Exemplos de imagens do *dataset*



Fonte: Elaborada pelos autores (2020).

Após a elaboração do *dataset*, implementou-se as arquiteturas de CNN, que é a etapa 2 da proposta. Para a codificação dos modelos, utilizou-se o *TensorFlow*⁴, definido como uma

⁴ Disponível em: <<https://www.tensorflow.org>>.

plataforma de código aberto para aprendizado de máquina. Além disso, também o *Keras*⁵ também foi usado, que é uma *API* de alto nível do *TensorFlow*, utilizada para criar e treinar modelos de aprendizado profundo. Em relação às CNNs escolhidas, este estudo optou pelas seguintes arquiteturas, a saber, VGG16, InceptionV3, DenseNet121, MobileNet e Xception. A escolha destas CNNs deu-se por serem as redes mais recentes do estado da arte na área, além de estarem disponíveis na *API Keras*. As arquiteturas foram implementadas usando duas técnicas em particular, a saber, aprendizagem por transferência, ou *transfer learning* em inglês, e ajuste fino, ou *fine tuning* em inglês. *Transfer learning* compreende a utilização de um modelo pré-treinado aplicado a um novo problema. Em resumo, utiliza-se uma rede neural pré-treinada para uma nova aplicação. Já o *fine tuning* reutiliza os pesos da rede original no novo modelo, e usando um novo conjunto de dados, efetua-se uma nova etapa de treinamento para adequar os pesos ao novo *dataset*. Neste trabalho, as camadas iniciais das CNNs pré-treinadas serviram como extrator de características. Em seguida, modificou-se o topo da rede para se adequar às classes do modelo e foi realizado um novo treinamento.

A etapa 3 relaciona-se com a tarefa de treinar e testar os modelos. A Figura 4 mostra a distribuição das imagens após a elaboração do *dataset*. Percebe-se que este encontra-se parcialmente equilibrado, apresentando uma média de 42 imagens por classe. Para treinar as arquiteturas, o conjunto de dados foi dividido usando a proporção de 70% para treino e 30% para teste. A Tabela 1 exibe a divisão das espécies de peixes por família e nome científico. Em resumo, analisou-se 4 tipos de famílias distribuídas em 8 tipos de peixes. É importante destacar que espécies da mesma família apresentam características muito semelhantes e o modelo tem o desafio de encontrar os atributos que diferenciam cada espécie.

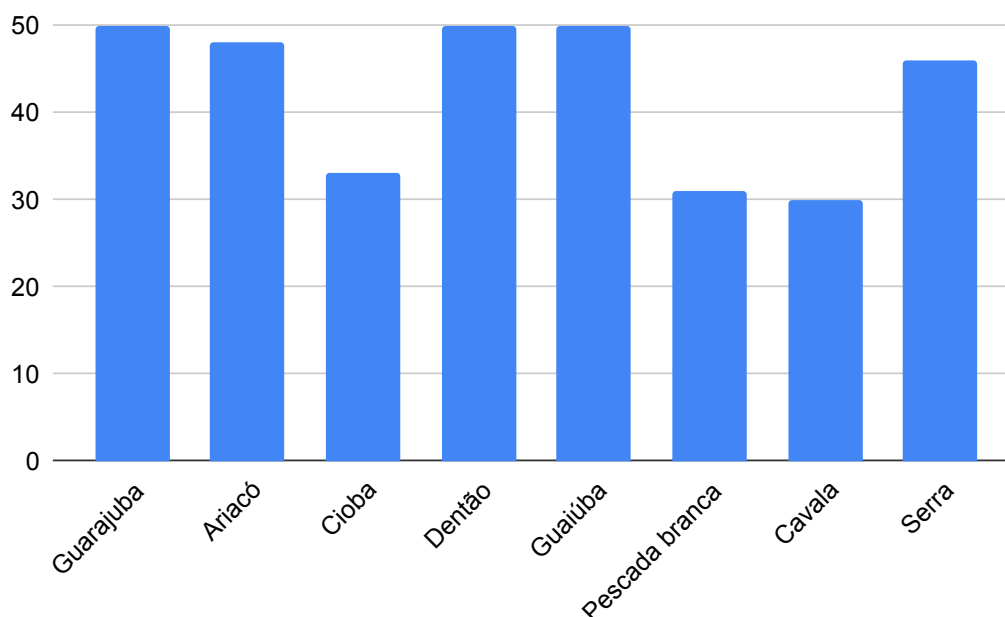
Tabela 1 – Distribuição das espécies por família

FAMÍLIA	ESPÉCIE (NOME POPULAR)	NOME CIENTÍFICO
<i>Carangidae</i>	Guarajuba	<i>Carangoides bartholomaei</i>
<i>Lutjanidae</i>	Ariacó	<i>Lutjanus synagris</i>
	Cioba	<i>Lutjanus analis</i>
	Dentão	<i>Lutjanus jocu</i>
	Guaiúba	<i>Lutjanus chrysurus</i>
<i>Sciaenidae</i>	Pescada branca	<i>Cynoscion leiarchus</i>
<i>Scombridae</i>	Cavala	<i>Scomberomorus cavalla</i>
	Serra	<i>Scomberomorus brasiliensis</i>

Fonte: Elaborada pelos autores (2020).

Com os modelos de CNN treinados e testados, chegou-se à etapa 4, a análise de resultados. Maiores detalhes dos resultados serão mostrados na Seção 5. Finalmente, na etapa 5, elaborou-se o *mockup* da aplicação *web* a partir do *software* Adobe XD para construção de protótipos. O *mockup* apresenta o *layout* da interface gráfica de um *software*. Ele é útil para demonstrar o uso e o resultado final de uma aplicação. Neste trabalho, construiu-se o protótipo de uma aplicação

⁵ Disponível em: <<https://keras.io/>>.

Figura 4 – Distribuição das imagens no *dataset*

Fonte: Elaborada pelos autores (2020).

web, visando a implementação futura a partir do modelo de CNN de melhor desempenho obtido nesta pesquisa.

5 ANÁLISE DOS RESULTADOS E DISCUSSÃO

A seguir, serão mostrados os resultados obtidos a partir da etapa de teste das CNNs.

Para obter os resultados mostrados nesta seção, elaborou-se dois cenários. O primeiro, no qual submeteu-se o *dataset* de teste durante as etapas de treinamento do topo da rede; e o segundo, no qual o conjunto de dados de teste foi submetido somente na etapa de teste do modelo. Os resultados alcançados para cada cenário são apresentados abaixo.

Considerando o cenário 1, a Figura 5 exibe as matrizes de confusão das CNNs. A matriz de confusão é uma tabela que permite visualizar as frequências de erros e acertos do modelo e é útil para analisar seu desempenho. Considerando a Figura 5a, que mostra a matriz da rede VGG16, observa-se que houve bom desempenho para as espécies Ariacó (13), Guaiúba (13), Dentão (14) e Guarajuba (17). Já para as espécies Cioba (4), Cavala (5) e Serra (5) houve uma baixa predição. A Figura 5b exibe o resultado para InceptionV3, tendo as espécies Serra (11), Guarajuba (12), Ariacó (13), Dentão (13) e Guaiúba (13) as maiores predições. Cavala (7) e Pescada Branca (7), por sua vez, obtiveram as menores predições. A matriz da DenseNet121 é exibida na Figura 5c. Nela, Ariacó (13), Guarajuba (14), Dentão (16) e Guaiúba (16) foram as espécies mais preditas corretamente, enquanto Cavala (7) e Cioba (7) novamente apresentaram desempenho inferior. A Figura 5d apresenta o resultado para MobileNet. As espécies Guarajuba (16), Serra (17) e Guaiúba (21) atingiram maior frequência de valores preditos corretamente. Nesta CNN, em particular, além das espécies Cioba (6) e Cavala

(7) apresentarem desempenho inferior, Dentão (9) também obteve o mesmo comportamento. Finalmente, a Figura 5e exibe os resultados para Xception. Nesta matriz, Guarajuba (15), Guaiúba (18) e Dentão (20) obtiveram as maiores frequências de valores preditos corretamente. Em oposição, Cioba (5) e Cavala (7) apresentaram baixas predições corretas. Sucintamente, as CNNs VGG16, InceptionV3 e DenseNet121 apresentaram melhores resultados considerando as matrizes de confusão. Concomitantemente, as espécies com maior quantidade de imagens no *dataset* apresentaram melhor desempenho, a saber, Ariacó, Dentão, Guaiúba e Guarajuba. Em contrapartida, as espécies Cavala e Cioba apresentaram desempenho inferior, fato este que pode estar relacionado à qualidade das imagens.

Ainda para o cenário 1, analisou-se as métricas Precisão, ou *precision* em inglês, Revocação, ou *recall* em inglês e *F-Score* em inglês, mostradas na Tabela 2. A Tabela 2a apresenta os resultados para a métrica *Precision*. A precisão indica a relação entre as predições positivas corretamente e todas as previsões positivas (incluindo as falsas). No geral, as espécies que obtiveram melhor desempenho independente da CNN foram Dentão, Guaiúba, Guarajuba e Pescada Branca. Em relação à arquitetura, o Xception, obteve melhor precisão com média de 0.89. Na Tabela 2b, exibe-se o desempenho para a métrica *Recall*. Esse parâmetro relaciona as previsões positivas realizadas corretamente e todas as previsões, ou seja, verdadeiros positivos e falsos negativos. Nesta tabela, as taxas oscilaram para todas as classes e arquiteturas. A CNN com melhor *Recall* foi MobileNet, com média de 0.85. A métrica *F-Score* é mostrada na Tabela 2c. Este padrão, por sua vez, é a média harmônica entre *precision* e *recall*, que indica a qualidade geral do modelo. Novamente, as taxas variaram para as classes e as CNNs, tendo a arquitetura Xception melhor desempenho, com média de 0.85. Em face do exposto, percebe-se a importância da quantidade de imagens por classe de forma a auxiliar no treinamento do modelo e, conseqüentemente, melhorar o desempenho deste ao ser submetido a testes.

Outro resultado obtido foi a acurácia das arquiteturas de CNN. A acurácia representa o quão próximo a predição está do resultado correto. Assim, quanto maior a acurácia, mais próximo do valor real é o resultado encontrado. Na Tabela 3 constata-se que Xception apresentou melhor acurácia, com taxa de 86%, assim como apresentou melhores resultados para as métricas *Precision* e *F-Score*.

Agora, serão apresentados os resultados para o cenário 2. O primeiro deles são as matrizes de confusão, mostradas na Figura 6. Ao considerar a arquitetura VGG16, têm-se as espécies Ariacó (13), Serra (14), Dentão (16) e Guarajuba (18) as maiores frequências de predições corretas, conforme mostra a Figura 6a. Cavala (4) e Cioba (6), entretanto, obtiveram as menores frequências de valores corretos. A Figura 6b exibe a matriz de confusão da InceptionV3. Nela, Serra (10), Ariacó (11), Dentão (11), Guaiúba (12) e Guarajuba (13) tiveram melhores resultados, enquanto que Cioba (6) novamente apresentou comportamento irregular. A matriz de confusão da DenseNet121 é exibida na Figura 5c. Para esta CNN, Dentão (15), Guaiúba (17) e Guarajuba (23) obtiveram mais predições corretas. Todavia, Ariacó (6), Cavala (6) e Cioba (6) tiveram desempenho inferior. A Figura 6d expõe os resultados para a arquitetura MobileNet. Neste resultado, as classes Ariacó (13), Guarajuba (14), Guaiúba (15) e Dentão (17) apresentam mais

Figura 5 – Matrizes de confusão para o cenário 1

(a) VGG16

Matriz de Confusão

Ariacó	13	0	0	0	0	0	0	0
Cavala	0	5	0	0	0	0	0	1
Cioba	7	0	4	0	0	1	0	0
Dentão	1	2	0	14	1	2	0	2
Guaiúba	3	0	1	0	13	1	0	1
Guarajuba	1	3	0	1	0	17	0	0
PescadaBranca	0	0	0	0	0	0	8	0
Serra	0	0	0	0	2	0	0	5

Classificação Correta

Classificação Preditada

(b) InceptionV3

Matriz de Confusão

Ariacó	13	0	0	0	3	0	0	0
Cavala	0	7	0	0	0	0	0	4
Cioba	0	0	10	1	2	0	0	0
Dentão	2	0	0	13	0	0	0	0
Guaiúba	0	0	1	0	13	0	0	0
Guarajuba	1	0	0	0	6	12	0	1
PescadaBranca	0	0	0	0	0	0	7	2
Serra	0	0	0	0	0	0	0	11

Classificação Correta

Classificação Preditada

(c) DenseNet121

Matriz de Confusão

Ariacó	13	0	0	0	0	0	0	0
Cavala	0	7	0	0	0	0	0	4
Cioba	3	0	7	1	0	0	0	0
Dentão	0	0	0	16	0	0	0	0
Guaiúba	4	0	2	0	16	0	0	1
Guarajuba	2	0	0	0	0	14	0	1
PescadaBranca	3	0	1	0	0	0	6	0
Serra	0	0	0	0	0	0	0	8

Classificação Correta

Classificação Preditada

(d) MobileNet

Matriz de Confusão

Ariacó	12	0	1	0	1	0	0	0
Cavala	0	7	0	0	0	1	0	2
Cioba	0	0	6	0	2	0	1	0
Dentão	1	0	0	9	0	0	0	0
Guaiúba	0	0	1	3	21	0	0	0
Guarajuba	0	0	0	0	1	16	1	1
PescadaBranca	0	0	0	0	0	0	5	0
Serra	0	0	0	0	0	0	0	17

Classificação Correta

Classificação Preditada

(e) Xception

Matriz de Confusão

Ariacó	8	0	0	0	3	0	0	0
Cavala	0	7	0	0	0	3	0	1
Cioba	0	0	5	1	2	0	0	0
Dentão	0	0	0	20	2	0	0	0
Guaiúba	0	0	0	1	18	0	0	0
Guarajuba	0	0	0	0	0	15	0	1
PescadaBranca	0	0	0	0	0	0	10	0
Serra	0	1	0	0	0	0	0	11

Classificação Correta

Classificação Preditada

Fonte: Elaborada pelos autores (2020).

Tabela 2 – Métricas para o cenário 1

(a) Métrica *Precision*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	0.52	0.50	0.80	0.93	0.81	0.81	1.00	0.56
InceptionV3	0.81	1.00	0.91	0.93	0.54	1.00	1.00	0.61
DenseNet121	0.52	1.00	0.70	0.94	1.00	1.00	1.00	0.57
MobileNet	0.92	1.00	0.75	0.75	0.84	0.94	0.71	0.85
Xception	1.00	0.88	1.00	0.91	0.72	0.83	1.00	0.85

(b) Métrica *Recall*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	1.00	0.83	0.33	0.64	0.68	0.77	1.00	0.71
InceptionV3	0.81	0.64	0.77	0.87	0.93	0.60	0.78	1.00
DenseNet121	1.00	0.64	0.64	1.00	0.70	0.82	0.60	1.00
MobileNet	0.86	0.70	0.67	0.90	0.84	0.84	1.00	1.00
Xception	0.73	0.64	0.62	0.91	0.95	0.94	1.00	0.92

(c) Métrica *F-Score*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	0.68	0.62	0.47	0.76	0.74	0.79	1.00	0.63
InceptionV3	0.81	0.78	0.83	0.90	0.68	0.75	0.88	0.76
DenseNet121	0.68	0.78	0.67	0.97	0.82	0.90	0.75	0.73
MobileNet	0.89	0.82	0.71	0.82	0.84	0.89	0.83	0.92
Xception	0.84	0.74	0.77	0.91	0.82	0.88	1.00	0.88

Fonte: Elaborada pelos autores (2020).

Tabela 3 – Métrica Acurácia para o cenário 1

	Acurácia
VGG16	0.72
InceptionV3	0.78
DenseNet121	0.79
MobileNet	0.85
Xception	0.86

Fonte: Elaborada pelos autores (2020).

predições positivas, enquanto que Cavala (8), Cioba (8) e Pescada Branca (8) resultaram em menos valores corretos. A matriz para a CNN Xception é apontada na Figura 6e, a qual mostra que Ariacó (12), Dentão (13), Guaiúba (13), Guarajuba (16) e Serra (16) obtiveram predições positivas mais frequentemente. Todavia, Cioba (5) e Cavala (6) apresentaram baixo desempenho. Diante do exposto, constata-se que as CNNs Xception e MobileNet apresentaram melhores desempenhos em termos de matriz de confusão. Novamente, é importante ressaltar que espécies com maior número de imagens tendem a ter melhores resultados.

Tabela 4 apresenta as métricas *Precision*, *Recall* e *F-Score* para o cenário 2. Considerando a métrica *Precision*, as espécies Dentão e Guaiúba obtiveram maiores taxas, conforme Tabela 4a. A CNN DenseNet121 apresentou melhor desempenho entre as classes, com média de 0.87. Ao se

Figura 6 – Matrizes de confusão para o cenário 2

(a) VGG16

Matriz de Confusão

Ariacó	13	0	2	0	2	0	0	0
Cavala	0	4	0	0	0	0	2	1
Cioba	0	0	6	0	2	0	0	0
Dentão	2	0	2	16	2	0	0	1
Guaiúba	0	0	0	2	9	0	0	0
Guarajuba	1	0	0	0	1	18	0	0
PescadaBranca	0	0	0	0	0	0	7	0
Serra	0	0	0	0	2	0	0	14

Classificação Correta

Classificação Predita

(b) InceptionV3

Matriz de Confusão

Ariacó	11	0	0	0	0	0	0	2
Cavala	0	8	0	0	0	0	0	2
Cioba	0	0	6	0	1	0	0	0
Dentão	0	1	5	11	0	0	0	0
Guaiúba	4	0	2	0	12	0	0	0
Guarajuba	0	2	0	1	1	13	0	0
PescadaBranca	0	2	0	0	0	0	9	0
Serra	0	3	0	0	1	1	1	10

Classificação Correta

Classificação Predita

(c) DenseNet121

Matriz de Confusão

Ariacó	6	0	0	1	4	0	0	0
Cavala	0	6	0	0	0	1	0	0
Cioba	0	0	6	1	1	0	0	0
Dentão	0	0	1	15	1	0	0	0
Guaiúba	0	0	0	1	17	0	0	0
Guarajuba	0	2	0	0	1	23	0	0
PescadaBranca	0	0	0	0	0	0	8	0
Serra	0	1	0	0	0	1	0	12

Classificação Correta

Classificação Predita

(d) MobileNet

Matriz de Confusão

Ariacó	13	0	0	0	1	0	0	0
Cavala	0	8	0	0	0	0	0	0
Cioba	0	0	8	0	2	0	0	0
Dentão	1	0	3	17	0	0	0	0
Guaiúba	0	0	0	0	15	1	0	0
Guarajuba	1	0	0	0	1	14	0	1
PescadaBranca	0	0	0	0	0	0	8	0
Serra	1	5	0	0	0	2	0	7

Classificação Correta

Classificação Predita

(e) Xception

Matriz de Confusão

Ariacó	12	0	0	0	3	0	0	0
Cavala	0	6	0	0	0	0	0	7
Cioba	0	0	5	0	0	0	0	0
Dentão	0	0	4	13	1	0	0	0
Guaiúba	0	0	6	0	13	0	0	0
Guarajuba	0	0	0	0	0	16	0	0
PescadaBranca	0	0	0	0	0	0	7	0
Serra	0	0	0	0	0	0	0	16

Classificação Correta

Classificação Predita

Fonte: Elaborada pelos autores (2020).

observar a métrica *Recall* na Tabela 4b, constata-se que todas as espécies tiveram taxas variadas para todas as CNNs. Novamente, a DenseNet121 obteve melhor eficácia com taxa média de 0.84. Já a métrica *F-Score*, exibida na Tabela 4c, evidencia que as classes Guarajuba e Pescada Branca obtiveram os melhores resultados. A DenseNet121 também obteve melhor resultado, com média de 0.83. Em suma, a arquitetura DenseNet121 apresentou melhor desempenho para as métricas da Tabela 4.

Tabela 4 – Métricas para o cenário 2

(a) Métrica *Precision*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	0.81	1.00	0.60	0.89	0.50	1.00	0.78	0.88
InceptionV3	0.73	0.50	0.46	0.92	0.80	0.93	0.90	0.71
DenseNet121	1.00	0.67	0.86	0.83	0.71	0.92	1.00	1.00
MobileNet	0.81	0.62	0.73	1.00	0.79	0.82	1.00	0.88
Xception	1.00	1.00	0.33	1.00	0.76	1.00	1.00	0.70

(b) Métrica *Recall*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	0.76	0.57	0.75	0.70	0.82	0.90	1.00	0.88
InceptionV3	0.85	0.80	0.86	0.65	0.67	0.76	0.82	0.62
DenseNet121	0.55	0.86	0.75	0.88	0.94	0.88	1.00	0.86
MobileNet	0.93	1.00	0.80	0.81	0.94	0.82	1.00	0.47
Xception	0.80	0.46	1.00	0.72	0.68	1.00	1.00	1.00

(c) Métrica *F-Score*

	Ariacó	Cavala	Cioba	Dentão	Guaiúba	Guarajuba	Pescada Branca	Serra
VGG16	0.79	0.73	0.67	0.78	0.62	0.95	0.88	0.88
InceptionV3	0.79	0.62	0.60	0.76	0.73	0.84	0.86	0.67
DenseNet121	0.71	0.75	0.80	0.86	0.81	0.90	1.00	0.92
MobileNet	0.87	0.76	0.76	0.89	0.86	0.82	1.00	0.61
Xception	0.89	0.63	0.50	0.84	0.72	1.00	1.00	0.82

Fonte: Elaborada pelos autores (2020).

A acurácia final dos modelos de CNNs é mostrada na Tabela 5. Observa-se que para o cenário 2, a arquitetura DenseNet121 apresentou melhor acurácia, com taxa de 85%, também demonstrando sua eficácia para as métricas analisadas anteriormente.

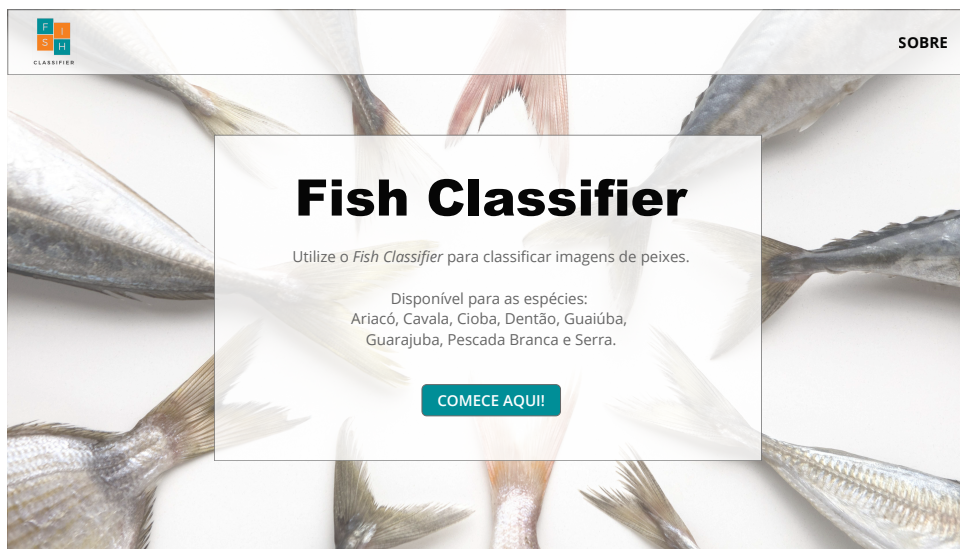
Tabela 5 – Métrica Acurácia para o cenário 2

	Acurácia
VGG16	0.79
InceptionV3	0.73
DenseNet121	0.85
MobileNet	0.82
Xception	0.80

Fonte: Elaborada pelos autores (2020).

Em síntese, percebe-se que, no cenário 1, a CNN Xception apresentou melhor desempenho com 86% de acurácia, e no cenário 2, a CNN DenseNet121 obteve desempenho de 85% em termos de acurácia. A diferença, em termos de acurácia para ambos os cenários, é mínima, apesar de, para cada cenário, ter havido uma CNN diferente.

Figura 7 – Tela inicial da aplicação *web*



Fonte: Elaborada pelos autores (2020).

Figura 8 – Tela de classificação da aplicação *web*



Fonte: Elaborada pelos autores (2020).

Após obter o modelo de CNN com melhor resultado a partir dos testes realizados, construiu-se o *mockup* para aplicação *web*, o qual será detalhado a seguir. Na tela inicial, apresentada na Figura 7, o usuário é apresentado ao sistema *Fish Classifier*. Optou-se por estabelecer um padrão de telas, que tem como ponto de partida a tela inicial. Um detalhe relevante é que a logo do sistema leva para a página inicial, em qualquer parte da aplicação. O botão “SOBRE”, por sua vez, redireciona para a tela com informações do *Fish Classifier*.

Figura 9 – Tela de resultado da aplicação *web*

Fonte: Elaborada pelos autores (2020).

Figura 10 – Tela Sobre da aplicação *web*

Fonte: Elaborada pelos autores (2020).

Ainda na tela inicial, tem-se um breve texto sobre a aplicação e as espécies disponíveis para classificação. O usuário pode realizar a classificação no botão “COMECE AQUI!”. A tela seguinte é a de classificação, exibida na Figura 8. Nela, o usuário pode carregar a imagem de peixe que deseja classificar. Em seguida, deve clicar no botão “CLASSIFICAR!”. Depois, o sistema redireciona para a página de resultado, mostrada na Figura 9. O usuário obtém o resultado do classificador, indicando a espécie a qual a imagem pertence. Caso ele deseje saber mais sobre a espécie, basta clicar na opção “Saiba mais sobre essa espécie”. Finalmente, na tela Sobre, exposta na Figura 10, é apresentado um texto sobre a aplicação com informações gerais, tais como descrição, objetivo, público-alvo, espécies disponíveis para classificação, entre outras. O usuário pode voltar para a tela inicial do sistema clicando na logo do *Fish Classifier*. Ao construir o *mockup*, objetivou-se mostrar um protótipo do sistema a ser implantado com o

modelo que apresentou melhor desempenho, neste caso, Xception com acurácia de 86%. Ao desenvolver a aplicação *web*, espera-se que esta tenha interface simples e, ao mesmo tempo, intuitiva para o usuário, conforme o protótipo mostrado nesta seção.

6 CONCLUSÃO E TRABALHOS FUTUROS

Este artigo apresentou um modelo de reconhecimento de peixes para prevenção de fraude em pescado. A partir do exposto, foi possível construir um *dataset* com imagens em ambientes reais. Ainda, após a implementação e comparação dos resultados das CNNs, constatou-se que a arquitetura Xception obteve melhor desempenho com 86% de acurácia. Também foi possível construir o *mockup* da aplicação *web*. Em face do exposto, conclui-se que o modelo pode ser aplicado para prevenção de fraude na cadeia de fornecimento de peixes.

Considerando as dificuldades para a implementação da proposta, têm-se a construção do conjunto de dados que é reduzido, mas que demandaria mais tempo para coletar mais imagens. Outro desafio foi a implementação das CNNs, desde a etapa de escolha delas até na análise dos resultados para entender o desempenho de cada arquitetura.

Os trabalhos futuros para a continuidade desta proposta baseiam-se no aumento da taxa de acurácia do modelo e na codificação da aplicação *web* a partir do *mockup* das telas. Assim, espera-se que o modelo possa contribuir para a prevenção da fraude em pescado, auxiliando sociedade, indústria e governo.

REFERÊNCIAS

- BÖHME, K. et al. Recent applications of omics-based technologies to main topics in food authentication. **TrAC Trends in Analytical Chemistry**, Elsevier, v. 110, p. 221–232, 2019.
- CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: **IEEE Conference on Computer Vision and Pattern Recognition**. Honolulu, HI, USA: IEEE, 2017. p. 1251–1258.
- HOWARD, A. G. et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. **arXiv preprint arXiv:1704.04861**, 2017.
- HRIDAYAMI, P.; PUTRA, I. K. G. D.; WIBAWA, K. S. Fish species recognition using vgg16 deep convolutional neural network. **Journal of Computing Science and Engineering**, v. 13, n. 3, p. 124–130, 2019.
- HUANG, G. et al. Densely connected convolutional networks. In: **IEEE Conference on Computer Vision and Pattern Recognition**. Honolulu, HI, USA: IEEE, 2017. p. 4700–4708.
- KAMILARIS, A.; PRENAFETA-BOLDÚ, F. X. Deep learning in agriculture: A survey. **Computers and Electronics in Agriculture**, Elsevier, v. 147, p. 70–90, 2018.
- KHAN, A. et al. A survey of the recent architectures of deep convolutional neural networks. **Artificial Intelligence Review**, Springer, p. 1–62, 2020.

KHASOGGI, B.; ERMATITA, E.; SAHMIN, S. Efficient mobilenet architecture as image recognition on mobile and embedded devices. **Indonesian Journal of Electrical Engineering and Computer Science**, v. 16, p. 389, 10 2019.

MASUDA, H.; JUKEI, T.; HASEGAWA, T. Fish species identification using a CNN-based multimodal learning method. In: **2nd International Conference on Image, Video and Signal Processing**. Jakarta, Indonesia: ACM, 2020. p. 15–19.

PROBST, W. N. How emerging data technologies can increase trust and transparency in fisheries. **ICES Journal of Marine Science**, Oxford Academy, v. 77, n. 4, p. 1286–1294, Jul.-Aug. 2020.

QIN, H. et al. Deepfish: Accurate underwater live fish recognition with a deep architecture. **Neurocomputing**, Elsevier, v. 187, p. 49–58, 2016.

RACHEL, J. J. L.; VARALAKSHMI, P. Classification of breeding fish using deep learning from the captured video. In: **11th International Conference on Advanced Computing**. Chennai, India: IEEE, 2019. p. 48–55.

RAUF, H. T. et al. Visual features based automated identification of fish species using deep convolutional neural networks. **Computers and Electronics in Agriculture**, Elsevier, v. 167, p. 105075, 2019.

ROSSI, F. et al. Fishapp: A mobile app to detect fish falsification through image processing and machine learning techniques. In: **IEEE International Conference on Automation, Quality and Testing, Robotics**. Cluj-Napoca, Romania: IEEE, 2016. p. 1–6.

SANGHVI, K. et al. Fauna image classification using convolutional neural network. **International Journal of Future Generation Communication and Networking**, v. 13, n. 1, p. 08–16, 2020.

SANTOS, A. A.; GONÇALVES, W. N. Improving pantanal fish species recognition through taxonomic ranks in convolutional neural networks. **Ecological Informatics**, Elsevier, v. 53, p. 100977, 2019.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.

SONG, W. et al. Use of smartphone videos and pattern recognition for food authentication. **Sensors and Actuators B: Chemical**, Elsevier, v. 304, p. 127247, 2020.

SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. In: **IEEE Conference on Computer Vision and Pattern Recognition**. Las Vegas, NV, USA: IEEE, 2016. p. 2818–2826.

VO, S. A. et al. Convolutional neural networks for individual identification in the southern rock lobster supply chain. **Food Control**, Elsevier, p. 107419, 2020.

WANG, D. et al. Modulation format recognition and osnr estimation using cnn-based deep learning. **IEEE Photonics Technology Letters**, IEEE, v. 29, n. 19, p. 1667–1670, 2017.

ZHOU, L. et al. Application of deep learning in food: A review. **Comprehensive Reviews in Food Science and Food Safety**, Wiley Online Library, v. 18, n. 6, p. 1793–1811, 2019.