Segmentação Semântica Multiclasse de Carcaças Bovinas

Antônio Almeida S. Neto[1],
Ricardo Ferreira[1], **José Augusto M. Nacif**[1]

[1]Instituto de Ciências Exatas e Tecnológicas – Universidade Federal de Viçosa (UFV)

{antonio.a.neto, ricardo, jnacif}@ufv.br

Resumo. *Devido a crescente competitividade da indústria e a necessidade cada vez maior em se produzir carne de qualidade, o uso de tecnologias como Inteligência Artificial e Redes Neurais Convolucionais na linha de produção torna-se indispensável, como por exemplo no contexto onde se é preciso extrair características para a avaliação de carcaças bovinas, como a delimitação dos cortes. O presente trabalho tem como objetivo a implementação de uma rede neural capaz de fazer a segmentação semântica da carcaça bovina, delimitando-a em 3 (três) classes de interesse: Dianteiro, Costelar e Traseiro. Através do desenvolvimento desta técnica, o processo de medição das carcaças se torna muito mais eficiente, diminuindo consideravelmente o tempo na linha de abate e a quantidade de interrupções, além de reduzir o contato dos profissionais envolvidos no processo com a carcaça, o que confere um risco menor de contaminação e mais confiabilidade ao processo. A principal métrica utilizada no trabalho foi a Perda de Entropia Cruzada, onde o modelo treinado obteve respectivamente 11% no conjunto de validação, o que representa um resultado satisfatório, principalmente levando em consideração o conjunto de dados composto por apenas 88 imagens, sendo 72 para treinamento e 16 para validação.*

1. Introdução

Pode-se afirmar que a produção de carne no Brasil é um dos pilares da agropecuária, o setor é de fato um dos carros-fortes da economia. Relevante para todo o mundo devido ao seu volume, qualidade e diversidade de produtos bovinos, suínos e de aves, o segmento representa um enorme desafio para as empresas que estão na sua linha de frente, buscando soluções modernas e formas de se destacar em relação aos concorrentes, potencializando lucro e trazendo processos inovadores para a indústria. Conforme [Muller 1987] as medidas dos cortes na carcaça de bovinos estão altamente correlacionadas com o seu peso e valor econômico sendo que, entre carcaças de medidas e acabamento similares, aquelas que possuem maior peso normalmente também contam com uma composição melhor e maior proporção da parte comestível em relação ao osso.

Os processos contidos na indústria da produção de carne têm sido amplamente explorados, no sentido de se otimizar e implementar soluções tecnológicas, como a utilização de sistemas de visão computacional que simulam o comportamento humano na execução das tarefas ligadas à linha de abate de bovinos, por exemplo. As Redes Neurais Convolucionais (RNCs) estão em ascensão no campo da visão computacional devido às suas vantagens: invariância de tradução, compartilhamento de parâmetros e conectividade esparsa [Yamashita et al. 2018]. Arquiteturas modernas de RNCs estão potencializando fortemente esses processos, principalmente através da segmentação de imagens,

cenário onde são capazes de atingir um nível muito alto de precisão na inferência, mesmo contando com um conjunto de dados limitado [Bonte et al. 2018].

Este trabalho propõe a implementação da U-Net [Ronneberger et al. 2015] para treinamento de um modelo capaz de fazer a segmentação de 3 (três) classes de corte da carcaça bovina, contribuindo no processo de medição e otimização da linha de abate. Através da segmentação automática das 3 classes (Traseiro, Costelar e Dianteiro), níveis satisfatórios na performance do modelo podem representar um ganho expressivo de eficiência na produção.

Muitos métodos de avaliação automática têm sido estudados para a extração de características relacionadas ao rendimento da carcaça bovina [Neto et al. 2019], e apesar das medições representarem uma parte muito importante na linha de abate, pouco se discute sobre o assunto em termos de visão computacional, essas tarefas normalmente são desempenhadas manualmente por um profissional técnico no frigorífico, trazendo muitas das vezes ineficiência e riscos de contaminação durante o processo.

Durante as seções seguintes deste trabalho, serão discutidos trabalhos mais recentes da literatura que buscam fazer a extração de características de carcaças bovinas, assim como a metodologia adotada no processo de obtenção do conjunto de dados utilizado no treinamento e a forma como esses dados foram introduzidos na rede. Ao final, serão apresentadas as métricas que foram utilizadas no desenvolvimento do trabalho, seus resultados e, por fim, algumas conclusões e possibilidades de trabalhos futuros.

2. Trabalhos Relacionados

Entre as tarefas mais exploradas no contexto da segmentação de imagens na literatura, pode-se dizer que a identificação e avaliação de animais têm ganhado destaque nos últimos anos, como pode ser verificado em [Gonçalves et al. 2020], onde foram utilizadas RCNs e o Superpixel, além de uma implementação da SegNet para fazer a segmentação de toda a região delimitada pela silhueta da carcaça bovina, produzindo um comparativo entre diferentes modelos de Redes Neurais Convolucionais e seus desempenhos respectivos na predição. Entre todos os modelos do comparativo, a arquitetura composta pela combinação do Superpixel e uma VGG16 obteve o melhor resultado, cerca de 92% no cálculo de IoU *(Intersection over Union)*.

Em [Daniel et al. 2020] também foram utilizadas técnicas de visão computacional em conjunto com um sistema de sensores, instalados em um frigorífico, de modo que fosse possível fazer a segmentação de toda a carcaça bovina em tempo real e em seguida a extração da concentração de gordura. Ao final do trabalho, é feita uma análise comparativa, considerando a classificação de 140 carcaças feita através do sistema e através da avaliação manual, a inferência obtida pelo sistema se mostra superior, com uma acurácia de 92,86%.

Ainda em [Lee et al. 2020], carcaças bovinas de Hanwoo (gado nativo da Coreia) tiveram seus pesos estimados através de um modelo desenvolvido com a utilização de 3 técnicas: Análise de Regressão Múltipla, Análise de Regressão de Mínimos Quadrados Parcial e Rede Neural Artificial (RNA), que foi submetido a um treinamento baseado em um conjunto de dados extraído de 134 carcaças Hanwoo. Através do cálculo de uma regressão linear, onde foi feito o comparativo entre os pesos preditos pelo modelo e os pesos reais, se obteve um $R^2 = 0,91$.

Apesar da recorrência de estudos relacionados à extração de características de carcaças bovinas, pode ser verificada uma concentração de trabalhos na literatura voltados para a obtenção dos níveis de gordura e peso da carcaça, de modo que informações associadas às medições não são comumente mapeadas no contexto de *machine learning* atualmente.

3. Metodologia

No decorrer desta seção, serão explanadas as principais atividades desempenhadas durante o desenvolvimento do trabalho, desde a obtenção e tratamento das imagens das carcaças bovinas, até a implementação da Rede Neural Convolucional que foi escolhida para treinar o modelo e segmentar as 3 (três) classes de corte.

3.1. Obtenção do Conjunto de Dados

O conjunto de dados que foi utilizado no trabalho é composto basicamente por imagens de carcaças bovinas e suas respectivas máscaras, sendo que as imagens correspondentes ao conjunto de validação não possuem máscaras. As imagens foram coletadas na resolução HD, dentro de um frigorífico local, onde estão instaladas 16 unidades da câmera *Intelbras IP 1230* ao longo de toda a linha de abate, entretanto, apenas as imagens obtidas pela filmagem de duas destas câmeras foram selecionadas, devido ao ângulo no qual às carcaças precisavam ser capturadas (vista lateral externa).

Ao todo, foram selecionadas 88 imagens para efetuar o treinamento e a validação, de modo que um especialista pudesse rotular suas regiões de interesse através do software VGG Image Annotator (VIA) [Dutta and Zisserman 2019]. A partir dos arquivos de anotações gerados, foi possível implementar um script em Python que recebe como entrada um arquivo .json contendo as coordenadas de cada região e gera máscaras em escala de cinza, onde cada classe é representada por uma cor diferente dentro da escala.

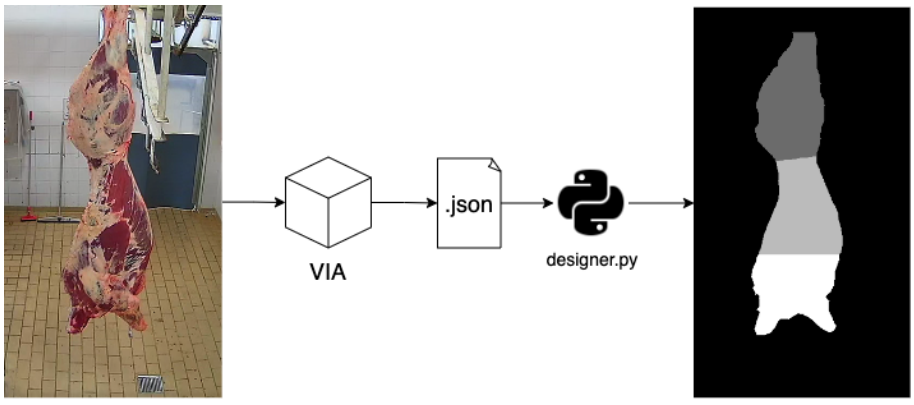

Figura 1. Processo até a geração das máscaras

Todos as imagens e máscaras foram armazenadas e organizadas na nuvem, com subpastas de acordo com as divisões para treino/validação e imagem/máscara, ou seja, 4 subdiretórios que foram utilizados como entrada no processo de treinamento do modelo. Através da implementação no Colaboratory foi feita a extração e processamento das imagens para persisti-las no ambiente de execução.

3.2. Arquitetura U-Net

O modelo implementada no trabalho foi baseado nas redes neurais convolucionais: comumente utilizadas no contexto de visão computacional e segmentação de imagens, as RNCs remontam aos anos 70, tendo em sua composição elementos de inspiração neural biológica, ideias como retropropagação, gradiente descendente, regularização, funções de ativação não lineares, entre outros estão presentes na arquitetura [LeCun et al. 2015].

A escolha da arquitetura veio principalmente devido ao seu nível excelente de desempenho em cenários onde o conjunto de dados não é tão expressivo, já que se tratava do contexto do trabalho, além da facilidade em se encontrar materiais de suporte na literatura e variações da sua implementação, compatíveis com os mais diversos tipos de ambientes e bibliotecas de aprendizado de máquina possíveis.

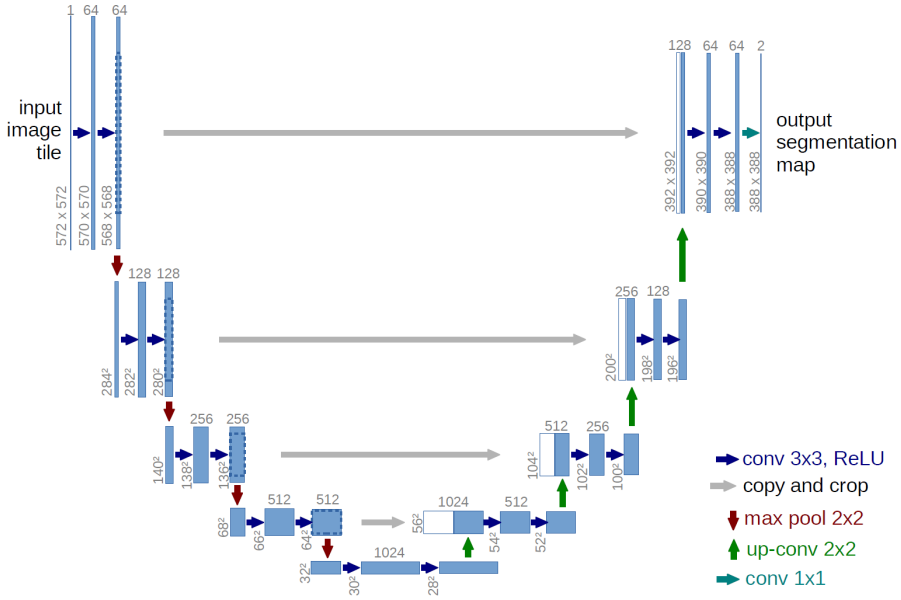

Figura 2. Arquitetura U-Net

A U-Net é uma rede totalmente convolucional, sua arquitetura propõe a utilização de camadas menos densas e uma quantidade reduzida de parâmetros, tendo como característica um processamento rápido e a possibilidade de se utilizar imagens de qualquer tamanho como entrada. Na fase inicial, é realizada uma redução da amostragem, fazendo o uso de convoluções, pooling e em seguida é feito o aumento da resolução, sendo que especificamente no caso da U-Net as etapas de aumento e redução da resolução também estão conectadas.

A arquitetura possui essa nomenclatura justamente devido ao seu formato na representação visual, que ilustra seu processamento passando basicamente por duas etapas: a de contração da imagem, fazendo a extração de características e a segunda, onde a resolução é aumentada por meio de deconvoluções, resultando ao final numa predição à nível de pixel.

3.3. Treinamento

Para a etapa de treinamento da rede, foi utilizado o conjunto de dados gerado após a rotulação pelo especialista, sendo que antes de fornecê-lo como entrada alguns ajustes

foram feitos de modo a otimizar a performance no processamento, como a remoção de obstruções nas imagens (pessoas e carcaças extras por exemplo) e o recorte de aproximadamente metade do conjunto, que estava na mesma resolução (HD ou 1280x720), porém numa orientação diferente (paisagem) da que foi definida para o restante das imagens.

Além disso, todas as imagens tiveram que ser redimensionadas devido à limitações de hardware, já que o tamanho original do conjunto era incompatível com a arquitetura utilizada, uma instância de máquina virtual pré-configurada fornecida pelo próprio ambiente de execução (Google Colaboratory). Apesar da necessidade de adaptação ao ambiente do Colaboratory, a redução no tamanho não foi tão significativa ao ponto de interferir no desempenho do treinamento.

Figura 3. Comparativo entre duas execuções

A codificação do treinamento foi feita à partir de uma implementação da UNet baseada em PyTorch, além da utilização do framework wandb *(Weights and Biases)*, que funciona como um ”TensorBoard Persistente”, ou seja, a ferramenta é capaz de armazenar os dados que são gerados durante o treinamento da rede neural, instanciando cada execução com seus parâmetros e resultados respectivos. Informações de entrada, como taxa de aprendizado, batch size, quantidade de épocas, entre outras são associadas à instância, juntamente com as métricas e resultados da mesma execução.

Através desta abordagem, a tarefa de fazer comparativos entre as execuções e modelos gerados fica muito mais simples e eficiente, tornando possível até mesmo o ajuste empírico de parâmetros que não são definidos de maneira trivial. O framework faz a coleta de todos os dados que são úteis durante a execução, e os disponibiliza por meio de sua plataforma com gráficos e representações visuais das predições que foram selecionadas através do código.

Na Figura 3, por exemplo, pode ser feito um comparativo entre a taxa de perda na etapa de validação entre um modelo que foi treinado por 20 épocas, com um conjunto de apenas 10% das imagens para validação, e outro modelo que foi treinado por 100 épocas com um *split* de 20% para validação. Nesse caso, o ajuste quantitativo no conjunto de imagens para validação foi decisivo para que a variação da perda fosse estabilizada, e a visualização comparativa ajudou muito no entendimento.

As execuções para validação do treinamento seguiram o protocolo de

experimentação *Holdout Cross-Validation* [Yadav and Shukla 2016], onde parte da base de dados é usada no treinamento, e um outro conjunto dos dados é utilizado especialmente para a etapa de validação, para que estes representem conjuntos diferentes entre si. Neste trabalho, inicialmente foram definidos valores arbitrários para os conjuntos, mas para o modelo final, 80% dos dados (imagens e máscaras) foram utilizados no conjunto de treinamento, enquanto 20% do *dataset* foi utilizado para realizar a validação.

3.4. Métricas

Para fazer uma avaliação quantitativa do método, foram utilizadas as métricas Perda de Entropia Cruzada e Precisão, sendo a primeira definida como prioritária e referência no processo de treinamento, de modo que o objetivo principal em relação aos ajustes no modelo fosse minimizar o valor da perda. Na teoria da informação, a entropia cruzada pode ser definida como sendo, basicamente, a diferença entre duas distribuições de probabilidade, que poderiam ser representadas no contexto deste trabalho como *p (máscaras)* e *q (predições)* sobre o conjunto de dados.

$$H(p, q) = -\sum_{x} p(x) \log q(x).$$

Figura 4. Fórmula da função de Perda de Entropia Cruzada

Através da utilização da métrica de Perda de Entropia Cruzada, cada predição gerada a partir do conjunto de validação é avaliada à nível de pixel em termos de pertencimento às classes definidas no modelo: Traseiro, Costelar, Dianteiro e *Background*, onde, para *pj* = 1 (máscara rotulada ou *ground truth*), e *qj* também igual a 1 (cenário de predição perfeita), o resultado da função de custo é 0, que é exatamente o que se procura, uma forma de medir o distanciamento entre as duas distribuições. Na próxima seção os resultados obtidos através da métrica proposta serão analisados.

Além da função de perda que foi utilizada como a métrica principal, a acurácia também foi mapeada durante as validações do modelo. A acurácia pode ser considerada uma das métricas mais simples e importantes, já que tem o papel de avaliar simplesmente o percentual de acertos, ou seja, seu cálculo pode ser definido basicamente pela razão entre a quantidade de acertos e o total de entradas, que neste contexto seria a média dos pixels que são inferidos corretamente em cada imagem do conjunto de validação, conjunto este que foi utilizado para medir a performance do modelo, mas também foram feitas medições de perda e acurácia sobre o conjunto de treinamento, principalmente para rastrear situações possíveis que deveriam ser evitadas, tais como o *overfitting* do modelo.

4. Resultados

Nesta seção, são apresentados os resultados obtidos durante a realização dos experimentos, onde muitas informações em relação ao modelo foram coletadas e avaliadas, principalmente levando em consideração as métricas estabelecidas na seção anterior e a própria avaliação visual das predições, outro ponto onde o framework wandb *(Weights and Biases)* contribuiu fortemente, já que foi possível, através da implementação, armazenar todas as predições do conjunto de validação no banco da ferramenta e fazer a plotagem de forma tabular, melhorando a eficiência nas comparações.

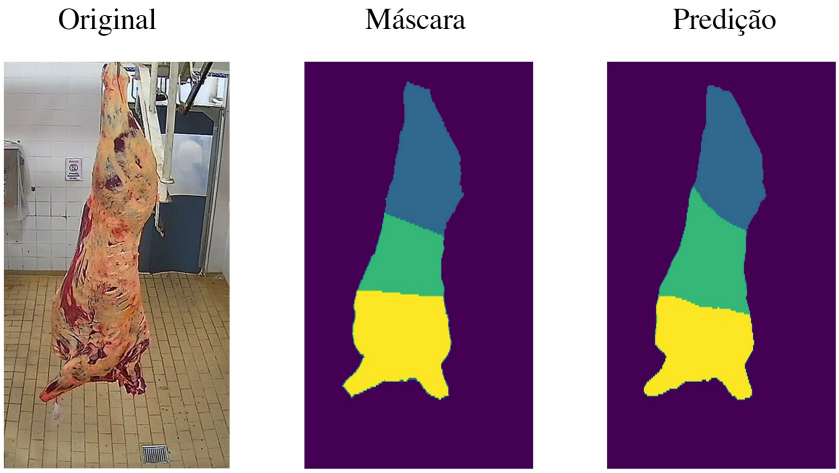

Figura 5. Predição do modelo final

Na Figura 5, pode-se verificar a predição de uma das amostras do conjunto de validação, sendo a primeira imagem (lado esquerdo) referente à imagem original da carcaça após o pré-processamento dos dados, a segunda (meio) é baseada na máscara rotulada pelo profissional dessa mesma carcaça, onde foi utilizado um script para converter o arquivo exportado pelo VIA em um bitmap e tornar possível a visualização *Ground Truth*. A terceira imagem (esquerda) representa a predição gerada pelo modelo, onde as classes estão definidas com a mesma cor de preenchimento, sendo: roxo para *Background*, azul para Traseiro, verde para Costelar e amarelo para Dianteiro.

Através da visualização das máscaras, foi possível constatar inconsistências no *dataset* e fazer ajustes para otimizar do modelo, problemas como regiões rotuladas incorretamente (ou até mesmo a ausência das mesmas) que poderiam afetar de forma significativa o processo de treinamento da rede, principalmente devido a limitação quantitativa do conjunto de dados. As predições se mostraram compatíveis com a Perda de Entropia Cruzada que foi obtida, tanto no conjunto de treinamento, quanto no conjunto de validação, cujos resultados respectivos são mostrados a seguir.

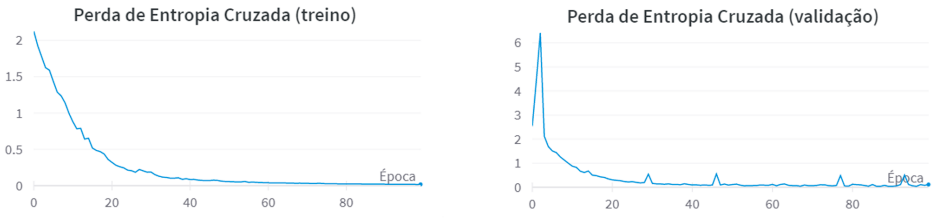

Figura 6. Resultados obtidos nos dois conjuntos de dados

Além da Perda de Entropia Cruzada, a acurácia obtida pelo modelo também foi mapeada, e apesar de ter uma função de métrica auxiliar neste trabalho, seus resultados se mostraram interessantes para os conjuntos de treinamento e validação, obtendo cerca de 98% e 96%, respectivamente. O cálculo da acurácia foi feito a partir da média entre as 4 (quatro) classes, ou seja, representa o total da área da carcaça como um todo que foi

segmentado corretamente, portanto, a métrica não faz uma análise individual por classe. Apesar dos números serem muito bons, a acurácia da carcaça como um todo pode não ser tão consistente na avaliação do modelo quanto a Perda de Entropia Cruzada, que faz uma análise considerando a perda de todas as classes, onde se encontra a maior preocupação do trabalho.

5. Conclusões e Trabalhos Futuros

A partir da visualização das predições e Perda de Entropia Cruzada que foram obtidas, pode-se dizer que o modelo se mostrou eficiente na segmentação das imagens das carcaças, principalmente levando em consideração a quantidade de imagens disponíveis para implementar o treinamento da rede, um conjunto relativamente pequeno, com 72 imagens para treinamento e 16 imagens para a etapa de validação.

Existem melhorias e avanços possíveis no trabalho, além da expansão do conjunto de dados, técnicas mais elaboradas de *augmentation* podem melhorar fortemente o desempenho da rede, além da própria captura das fotos na linha de abate, onde um posicionamento mais adequado pode tornar possível a utilização de mais câmeras, já que neste trabalho foram utilizados frames de apenas 2 câmeras das 16 disponíveis no frigorífico, devido ao ângulo desfavorável para pegar a vista lateral externa das carcaças.

Além das possibilidades de melhoria, o presente trabalho pode ser estendido num processo de medição dos cortes, através da utilização das predições com o auxílio de alguma biblioteca de visão computacional, por exemplo, de modo que a representação visual gerada seja fornecida na entrada e as medidas sejam extraídas de acordo com o contorno das regiões geradas pelo modelo.

Referências

Bonte, S., Goethals, I., and Van Holen, R. (2018). Machine learning based brain tumour segmentation on limited data using local texture and abnormality. *Computers in biology and medicine*, 98:39–47.

Daniel, H., González, G. V., García, M. V., Rivero, A. J. L., and De Paz, J. F. (2020). Non-invasive automatic beef carcass classification based on sensor network and image analysis. *Future Generation Computer Systems*, 113:318–328.

Dutta, A. and Zisserman, A. (2019). The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*. ACM.

Gonçalves, D. N., de Moares Weber, V. A., Pistori, J. G. B., da Costa Gomes, R., de Araujo, A. V., Pereira, M. F., Gonçalves, W. N., and Pistori, H. (2020). Carcass image segmentation using cnn-based methods. *Information Processing in Agriculture*.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Lee, D.-H., Lee, S.-H., Cho, B.-K., Wakholi, C., Seo, Y.-W., Cho, S.-H., Kang, T.-H., and Lee, W.-H. (2020). Estimation of carcass weight of hanwoo (korean native cattle) as a function of body measurements using statistical models and a neural network. *Asian-Australasian Journal of Animal Sciences*, 33(10):1633.

Muller, L. (1987). Normas para avaliação de carcaças e concurso de carcaças de novilhos. *UFSM Santa Maria, Imprensa Universitária*, page 31.

Neto, A. B., Bonini, C., Putti, F., Campos, M., Gabriel Filho, L., Chacur, M., and Piazentin, J. (2019). Modelo automático de classificação de bovinos para o abate via redes neurais artificiais. *Revista Brasileira de Engenharia de Biossistemas*, 13(1):1–11.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Yadav, S. and Shukla, S. (2016). Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In *2016 IEEE 6th International conference on advanced computing (IACC)*, pages 78–83. IEEE.

Yamashita, R., Nishio, M., Do, R. K. G., and Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9(4):611–629.