

# A Weight Monitoring Platform for Small Ruminants Using U-Net and EfficientNetB1

Rafael L. Araújo<sup>1,2</sup>, Júlio V. M. Marques<sup>1</sup>, Anthony I. M. Luz<sup>1</sup>,  
Marina C. M. Santos<sup>1</sup>, João dos Santos Neto<sup>1</sup>, Alcilene D. de Sousa<sup>1</sup>,  
Frank C. L. Veras<sup>1</sup>, Antônio O. de Carvalho Filho<sup>1</sup>, Romuere R. V. e Silva<sup>1</sup>

<sup>1</sup>Universidade Federal do Piauí (UFPI) – Picos – PI – Brasil

<sup>2</sup>Instituto Federal do Piauí (IFPI) – Picos – PI – Brasil

rafaluzaraujo@ifpi.edu.br, romuere@ufpi.edu.br

**Abstract.** *The raising of goats and sheep on small farms in Brazil faces challenges in monitoring animal weight due to the lack of precise scales, which are essential for proper nutritional and health management. This study proposes a web-based system for the automatic and non-invasive estimation of small ruminants' body weight. Developed with open-source technologies such as Django and SQLite, the system employs deep learning with U-Net for animal segmentation, EfficientNetB1 for feature extraction, and regression models for weight prediction. A total of 203 images from 42 animals were evaluated, with promising results: Dice score of 97.7 for segmentation, MAE of 2.65, RMSE of 4.20, and R<sup>2</sup> of 0.81 for regression. The solution offers competitive performance and low cost, made possible by using a standard camera and free software.*

**Resumo.** *A criação de caprinos e ovinos em pequenas fazendas no Brasil enfrenta desafios no monitoramento do peso dos animais, devido à ausência de balanças precisas, essenciais para o manejo nutricional e sanitário. Este estudo propõe um sistema web para estimativa automática e não invasiva do peso corporal de pequenos ruminantes. Desenvolvido com tecnologias de código aberto, como Django e SQLite, o sistema utiliza deep learning com U-Net para segmentação dos animais, EfficientNetB1 para extração de características e regressores para a predição de peso. Foram avaliadas 203 imagens de 42 animais, com resultados promissores: Dice de 97,7 na segmentação, MAE de 2,65, RMSE de 4,20 e R<sup>2</sup> de 0,81 na regressão. A solução apresenta desempenho competitivo e baixo custo, viabilizado pelo uso de câmera comum e software livre.*

## 1. Introdução

O Brasil é um país de expressivo rebanho bovino, contudo, apresenta baixa eficiência produtiva comparada a dos Estados Unidos que, em 2021, lideraram a produção global de carne bovina, atingindo 12,7 milhões de toneladas de equivalente carcaça, contando com aproximadamente 92 milhões de animais. Em contraste, o Brasil ficou em segundo lugar, registrando cerca de 9,7 milhões [ABIEC 2022]. Além do rebanho bovino, o Brasil também detém uma significativa presença na criação de pequenos ruminantes, contabilizando aproximadamente 11,3 milhões de caprinos e 19,7 milhões de ovinos em 2019 [of Geography and Statistics 2018]. Essas espécies, categorizadas como pequenos ruminantes, desempenham um papel crucial na ovinocaprinocultura,

setor que, embora ainda não tenha atingido as grandezas observadas na bovinocultura nacional, apresenta um expressivo aumento na comercialização ao longo dos anos [NÓBREGA and VERGNE 2018, DROVERS 2020].

Diante do potencial de crescimento deste segmento, torna-se imperativo aprimorar técnicas modernas de manejo, incluindo programas de melhoramento genético e o monitoramento contínuo do rebanho [de LUCENA and GUIMARAES 2018]. Nas raças destinadas ao corte, a medição precisa das características corporais se revela essencial para a estimativa do peso e a classificação de fenótipos, contribuindo para a criação de animais mais saudáveis e produtivos [Sabbioni et al. 2020]. A pesagem de animais em fazendas enfrenta desafios significativos, desde a escassez de balanças em condições de campo até os elevados custos associados e a vulnerabilidade dos equipamentos eletrônicos a danos [Martins et al. 2020, Song et al. 2018].

Nesse contexto, a predição do peso corporal por meio de medidas morfológicas, como a circunferência torácica, altura da cernelha e largura do quadril, emerge como uma alternativa viável [Dickinson et al. 2013]. Entretanto, essas medidas frequentemente são subjetivas, demandam trabalho extenso e podem gerar estresse nos animais [Dickinson et al. 2013]. Além disso, o aumento da preocupação com o bem-estar animal e a crescente demanda por práticas sustentáveis evidenciam a necessidade de abordagens não invasivas e respeitosas ao rebanho [Zhang et al. 2018]. Surge, assim, diversas propostas baseadas em sistema de visão computacional para pesagem automática de ovinos e caprinos, oferecendo vantagens como custo reduzido, durabilidade e uma abordagem não intrusiva por meio de câmeras estrategicamente posicionadas [Liu et al. 2020].

Com base no exposto, este projeto visa a criação de novas metodologias baseadas em *deep learning* para estimar o peso corporal desses animais a partir de imagens, alinhando-se às exigências contemporâneas por métodos eficientes e éticos na pecuária. O objetivo deste trabalho é projetar e avaliar uma arquitetura baseada em modelos pré-treinados de segmentação de imagens e regressão, voltada à predição do peso de pequenos ruminantes, por meio de um sistema web. A proposta utiliza a rede U-Net para a segmentação das imagens e CNNs pré-treinadas com regressão para estimativa do peso, permitindo o monitoramento contínuo e automatizado dos animais, e contribuindo para uma gestão mais eficiente e sustentável dos rebanhos.

## 2. Trabalhos Relacionados

Ao revisar o estado da arte, identificam-se diversos estudos que abordam a estimativa do peso corporal de pequenos ruminantes por meio de técnicas de visão computacional e aprendizado de máquina. Apesar dos avanços relevantes observados nessas abordagens, torna-se essencial analisar suas limitações, a fim de direcionar o desenvolvimento de pesquisas futuras e propor soluções mais eficazes.

O estudo de [Samperio et al. 2021] empregou 100 imagens tridimensionais para estimar o peso de cordeiros, alcançando um coeficiente de determinação ( $R^2$ ) de 86%. Contudo, a pesquisa não detalha métricas de erro como MAE ou RMSE, dificultando a avaliação completa da precisão do modelo. Além disso, a necessidade de equipamentos especializados para captura de imagens 3D pode limitar a aplicabilidade em pequenas propriedades rurais. [Sant'Ana et al. 2021] combinaram técnicas de visão computacional com algoritmos de regressão para pesar ovelhas vivas, obtendo um MAE de 3,099 kg e

RMSE de 3,481 kg com base em 150 imagens. Apesar dos resultados promissores, o coeficiente de determinação médio foi de 68,7%, indicando que aproximadamente 31,3% da variabilidade do peso não foi explicada pelo modelo. Isso sugere a necessidade de incluir variáveis adicionais ou aprimorar as técnicas de modelagem para aumentar a precisão.

[He et al. 2023] utilizaram a arquitetura *LiteHRNet* aprimorada com 80 imagens RGB-D para estimar o peso de ovelhas, alcançando um MAE de 6,736 kg e RMSE de 8,866 kg. Esses valores indicam uma margem de erro considerável, possivelmente devido à complexidade na integração de informações de cor e profundidade ou limitações na qualidade dos dados coletados.

[Menesatti et al. 2014] aplicaram visão computacional estereoscópica com 120 imagens 2D para estimar o peso de ovinos, obtendo um MAE de 3,54 kg. Embora o erro médio absoluto seja relativamente baixo, a pesquisa não apresenta outras métricas de desempenho ou detalhes sobre a variabilidade dos dados, dificultando uma avaliação mais abrangente da eficácia do método. Por fim, [Oliveira et al. 2021] focaram na segmentação de 50 imagens de ultrassonografia de carcaça de pequenos ruminantes usando aprendizado profundo. Apesar de avanços na segmentação, o estudo não aborda diretamente a estimativa de peso corporal, limitando sua aplicabilidade para esse fim específico.

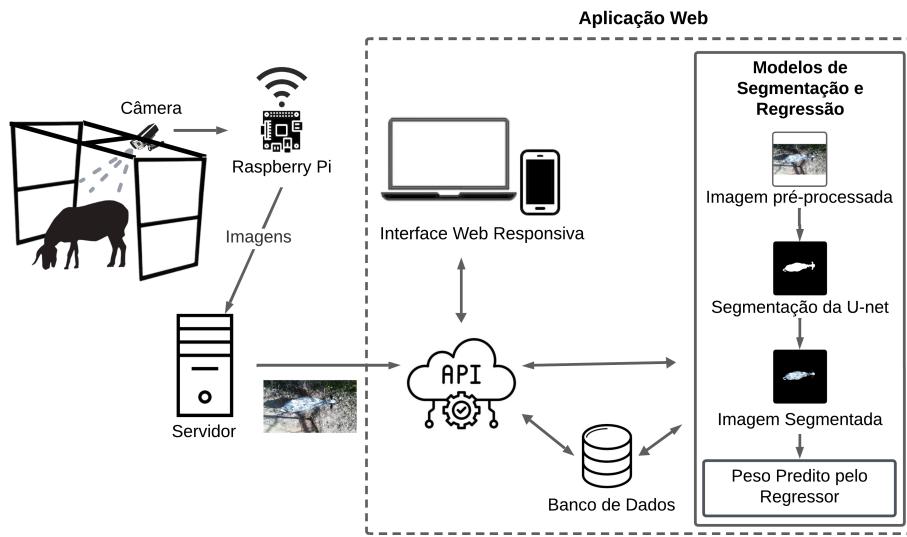
Essas limitações evidenciam a necessidade de desenvolver modelos mais robustos e adaptáveis, que considerem a variabilidade dos dados e sejam viáveis para implementação em diferentes contextos de produção animal. A integração de variáveis adicionais, aprimoramento das técnicas de modelagem e consideração das condições práticas das propriedades rurais são aspectos cruciais para futuras pesquisas na área. Neste contexto, o presente trabalho se destaca por propor uma abordagem integrada de segmentação e regressão com redes convolucionais pré-treinadas, aliada ao desenvolvimento de uma plataforma Web funcional, voltada à aplicação prática na rotina de criadores. O sistema foi treinado com dados reais de animais capturados em ambiente de criação, utilizando tecnologias acessíveis com foco na futura adoção por pequenos e médios produtores.

### **3. Metodologia**

A metodologia proposta nesta pesquisa tem por objetivo a criação de um sistema Web para estimação de peso de ovinos e caprinos de forma automática e segue o fluxo presente na Figura 1. Partindo da aquisição das imagens que são enviadas para o sistema através da API, em seguida essas imagens servem de entrada para modelos de segmentação e regressão que farão a predição do peso e os dados são salvos no banco de dados para visualização através da interface Web responsiva.

#### **3.1. Aquisição de Imagens**

A aquisição de imagens é uma etapa crucial no processo de estimativa de peso dos pequenos ruminantes. A infraestrutura desenvolvida para capturar essas imagens é composta por uma câmera Kinect v2, da Microsoft, conectada a um microcontrolador (*Raspberry Pi*) e instalada em uma estrutura física projetada para garantir uma visão superior completa do corpo dos animais. A câmera utiliza tecnologia *Time of Flight* e possui resolução de 1920×1080 pixels para imagens coloridas, operando a 30 quadros por segundo. O posicionamento superior foi escolhido para capturar a silhueta dos ovinos e caprinos de forma padronizada, favorecendo a etapa de segmentação.



**Figura 1.** Arquitetura geral do sistema proposto.

A estrutura é construída com materiais resistentes e posicionada de modo que os animais transitem naturalmente por ela. Atualmente, é necessária a condução manual dos animais até a estrutura, e a captura das imagens é acionada por um operador por meio de um notebook conectado ao microcontrolador. As imagens capturadas são enviadas para um servidor central, permitindo sua sincronização com o sistema por meio de uma API. A base de dados utilizada neste trabalho é composta por 203 imagens de 42 ovinos e caprinos.

Essa abordagem ainda é semi-automatizada, porém pretende-se automatizar a captura das imagens através de sensores de movimento gerenciados pelo microcontrolador, dessa forma removendo a interação humana para manejo dos animais e captura das fotos. Além disso, o uso de tecnologias acessíveis e de baixo custo, como o *Raspberry Pi*, torna o sistema viável para diferentes tipos de propriedades, desde pequenas fazendas até operações de grande escala.

### 3.2. Segmentação de Imagens com U-net

A *U-Net*, proposta originalmente por [Ronneberger et al. 2015], é uma rede neural totalmente convolucional projetada para a segmentação de imagens biomédicas. Sua arquitetura consiste em um caminho de contração que captura informações contextuais, seguido por um caminho de expansão simétrico que permite uma segmentação precisa. O diferencial da *U-Net* em relação a outras arquiteturas de segmentação reside na combinação entre os mapas de características do estágio de contração com seus correspondentes no estágio de expansão. Essa abordagem facilita a propagação de informações contextuais para os mapas de características de alta resolução, aprimorando a precisão da segmentação. Por ser pré-treinada [Iakubovskii 2019], a quantidade de camadas difere da original, variando de acordo com o *backbone* escolhido. Neste trabalho, utilizamos a ResNet34 devido ao desempenho obtido em experimentos preliminares, nos quais alcançamos um Dice de 92% na segmentação do *dataset* PH2 [Araújo et al. 2022].

### **3.3. Regressão para Estimação de Peso**

Para a extração de características das imagens capturadas, utilizamos as seguintes arquiteturas: *Xception*, *VGG19*, *InceptionResNetV2* e *EfficientNetB1*. Essas arquiteturas foram escolhidas devido ao seu desempenho superior em tarefas de classificação e extração de características, aproveitando a profundidade e a complexidade dos modelos para capturar informações detalhadas das imagens [Chollet 2017, Simonyan and Zisserman 2014, Szegedy et al. 2017, Tan and Le 2019].

As características extraídas por cada rede foram utilizadas como entrada para diferentes modelos de regressão: Linear Regression, Random Forest, SVR, Lasso, ElasticNet, Ridge, Gradient Boosting e KNeighbors. Cada combinação de arquitetura e regressor foi avaliada separadamente, de forma independente, com o objetivo de identificar o par com melhor desempenho. Essa abordagem possibilitou a criação de modelos eficientes para a estimativa de peso a partir de imagens, promovendo uma solução não invasiva e precisa para o monitoramento dos animais [Pedregosa et al. 2011].

### **3.4. Desenvolvimento do Sistema Web**

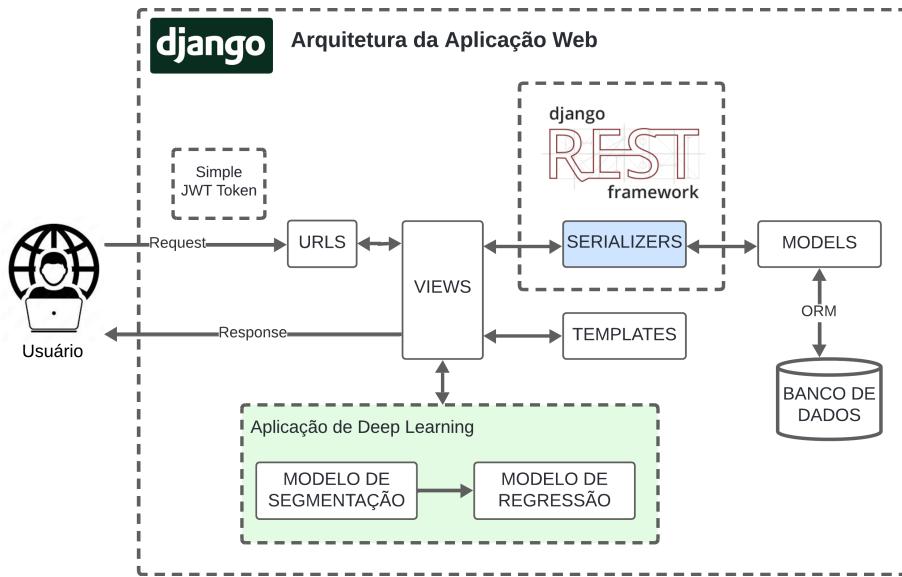
Para a construção do sistema, utilizou-se uma combinação de tecnologias robustas e amplamente adotadas. O framework *Django* foi escolhido para o desenvolvimento do *backend* devido à sua facilidade de uso, segurança e capacidade de escalabilidade [Vincent 2018]. *Django Rest Framework (DRF)* foi empregado para a construção da API, proporcionando uma interface eficiente para a comunicação entre o *frontend* e o *backend* [Christie and Gibson 2024]. Para a autenticação na API, utilizamos o *Simple JWT*, garantindo a segurança das operações dentro do sistema [Allen et al. 2024].

O banco de dados utilizado foi o *SQLite*, uma escolha prática para o desenvolvimento inicial devido à sua simplicidade e facilidade de integração com *Django* [Kreibich 2010]. No entanto, o sistema foi projetado para permitir a migração para bancos de dados mais robustos, como *PostgreSQL* ou *MySQL*, caso a necessidade de escalabilidade aumente. Para a criação e implementação dos modelos utilizados na predição de peso, empregamos bibliotecas como *TensorFlow* e *Keras* para a construção e treinamento dos modelos de *deep learning* para extração de características [Abadi et al. 2016, Chollet et al. 2015] e *Scikit-learn* para algoritmos de *machine learning* aplicados na tarefa de regressão [Pedregosa et al. 2011].

#### **3.4.1. Arquitetura do Sistema**

O usuário, por meio do navegador web, envia requisições para as rotas definidas nos arquivos `urls.py`. Cada rota aciona uma view, responsável pela lógica da aplicação e pela chamada dos modelos de deep learning. As rotas foram organizadas para refletir os principais módulos do sistema, como envio de imagens, segmentação, predição de peso e visualização de resultados. Nos arquivos `models.py`, definem-se classes que representam as entidades persistidas no banco de dados SQLite por meio do Object Relational Mapper (ORM). Uma ilustração da arquitetura do sistema é apresentada na Figura 2.

A API foi criada para o recebimento remoto das imagens coletadas em campo, através de arquivos *serializers* os dados são serializados para o formato *json*, garantindo



**Figura 2. Arquitetura interna da aplicação Web baseada em Django.**

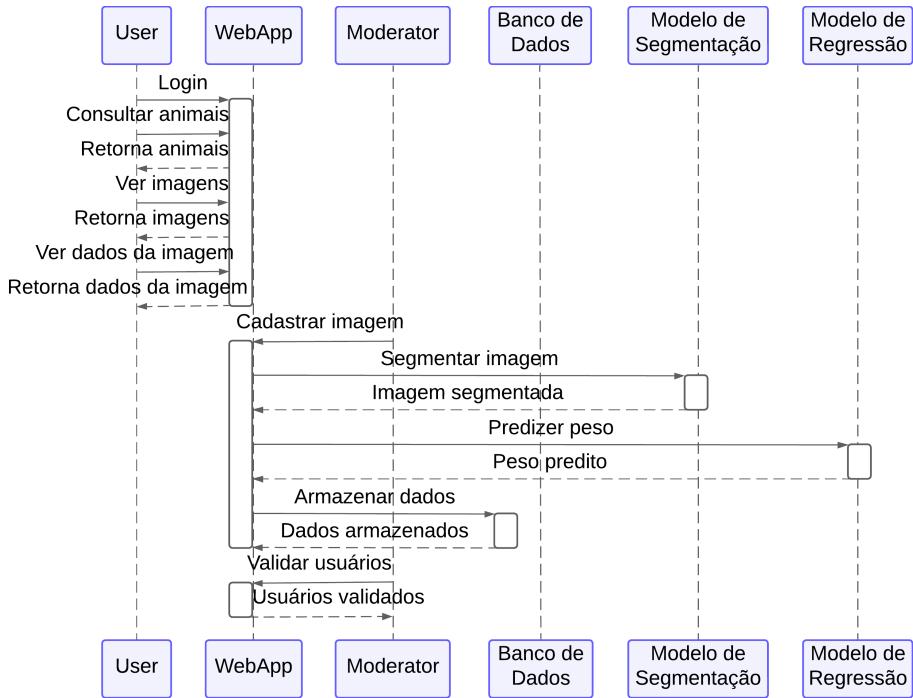
sua fácil obtenção, manipulação e distribuição. Para garantir que apenas pessoas autorizadas utilizem as rotas da API utilizou-se o *Simple JWT*, onde o usuário deve acessar uma rota enviando suas credenciais para geração de um *token* de acesso, que possui tempo de expiração definido pelo sistema.

### 3.4.2. Funcionalidades do Sistema

O sistema Web desenvolvido conta com funcionalidades específicas para diferentes perfis de usuários, como cadastro e aprovação de novos usuários, login, registro de animais, upload de imagens e visualização de dados. Moderadores podem ainda processar as imagens por meio dos modelos de segmentação e regressão, além de cadastrar informações adicionais relevantes. O fluxo geral dessas operações é ilustrado na Figura 3. Inicialmente, o usuário acessa o sistema e realiza o login, que consulta o banco de dados para autenticação. Em seguida, o usuário pode solicitar a listagem de animais e visualizar as imagens associadas, com os dados sendo recuperados do banco de dados. Novas imagens podem ser enviadas, sendo armazenadas e processadas pelos modelos de segmentação e predição de peso. Os resultados são então salvos no banco de dados. Por fim, moderadores podem aprovar novos usuários, atualizando seus status no sistema.

### 3.4.3. Interface do Usuário

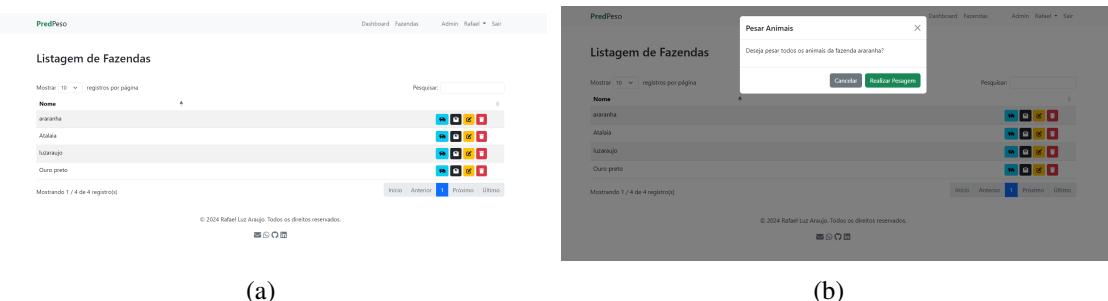
A interface do usuário do sistema foi projetada para ser intuitiva e funcional, facilitando o uso por parte de criadores de animais e administradores. Durante o desenvolvimento, foram seguidos princípios básicos de usabilidade e boas práticas de *User Experience (UX)*, como a consistência visual, hierarquia clara de informações, feedback ao usuário e uso de componentes familiares de interface. As principais telas do sistema incluem a tela de login, que é a primeira interface acessada e permite a autenticação segura com credenciais personalizadas, e o *dashboard*, que apresenta uma visão geral com métricas e informações



**Figura 3. Diagrama de Sequência.**

relevantes sobre os animais.

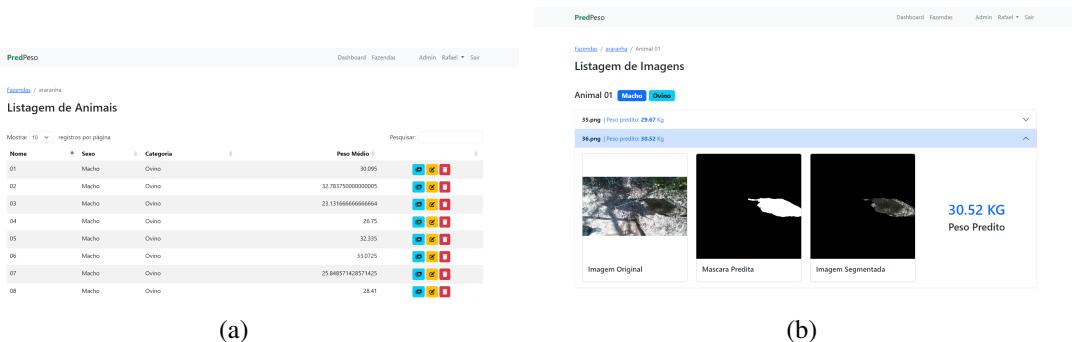
Na Tela de Fazendas (Figura 4a), o usuário pode gerenciar diferentes fazendas cadastradas no sistema. Aqui, é possível visualizar a lista de fazendas, adicionar novas fazendas, e editar ou remover fazendas existentes. A (Figura 4b) apresenta a funcionalidade de pesagem que permite ao usuário iniciar o processo de pesagem dos animais de uma fazenda específica. Através desta tela, é possível enviar imagens para segmentação e predição de peso, além de visualizar os resultados das pesagens anteriores.



**Figura 4. Telas do Sistema. (a) Fazendas, (b) Pesagem.**

A Tela de Animais (Figura 5a) exibe uma lista de todos os animais cadastrados no sistema para uma determinada fazenda, permitindo que o usuário adicione novos animais ou edite informações dos animais existentes. Por fim, a Tela de Imagens do Animal (Figura 5b) mostra todas as imagens associadas a um animal específico. Nessa tela, o usuário pode enviar novas imagens para segmentação e predição, além de visualizar os resultados das análises anteriores. Atualmente, a identificação dos animais é realizada manualmente por meio de um identificador único atribuído no momento do cadastro.

Como trabalho futuro, pretende-se incorporar mecanismos automáticos de identificação individual dos animais, como o uso de reconhecimento facial ou de pelagem.



**Figura 5. Telas do Sistema. (a) Animais, (b) Imagens do Animal.**

## 4. Resultados e Discussão

Serão apresentados os resultados obtidos nos testes de segmentação e regressão. Os testes foram realizados utilizando diferentes modelos e parâmetros para avaliar a performance dos algoritmos.

### 4.1. Avaliação da Segmentação de Imagens

Para segmentação utilizou-se a *U-net* pré-treinada, com a arquitetura de *backbone ResNet34*. Testes prévios definiram hiperparâmetros como otimizador, taxa de aprendizagem, entre outros. Em seguida, foram realizados testes utilizando diferentes tamanhos de imagens, cujos resultados estão apresentados na Tabela 1.

**Tabela 1. Resultados obtidos em diferentes tamanhos de imagem para a U-net pré-treinada.**

Image size	Sensitivity (%)	Specificity (%)	Accuracy (%)	AUC (%)	Dice (%)	Jaccard (%)
128x128	94.57	99.74	99.47	97.16	94.97	90.42
224x224	95.55	99.86	99.63	97.70	96.49	93.22
512x512	<b>97.29</b>	<b>99.90</b>	<b>99.76</b>	<b>98.59</b>	<b>97.70</b>	<b>95.50</b>

Os resultados mostram que o tamanho de imagem  $512 \times 512$  apresentou os melhores resultados em todas as métricas avaliadas, incluindo *Sensitivity*, *Specificity*, *Accuracy*, *AUC*, *Dice* e *Jaccard*. Isso pode ser explicado pelo fato de que imagens maiores contêm mais informações detalhadas, permitindo ao modelo capturar melhor as características relevantes para a segmentação. A maior resolução das imagens pode ter contribuído para uma segmentação mais precisa e detalhada, resultando em melhores métricas de performance.

### 4.2. Desempenho dos Modelos de Regressão

As regiões segmentadas com a *U-Net* são utilizadas como base para a extração de características morfológicas e estruturais por meio de arquiteturas convolucionais pré-treinadas. Essas características, por sua vez, são empregadas como entrada para modelos de regressão, responsáveis por estimar o peso corporal dos animais a partir das imagens.

Inicialmente, foram realizados testes com 42 animais, obtendo-se resultados promissores. Contudo, observou-se que a presença de valores discrepantes prejudicava o desempenho. Os pesos dos animais variavam de 6.8 kg a 62.1 kg, com a maioria abaixo de 44.2 kg e apenas um animal com 62.1 kg.

Conforme [Klivans et al. 2018], os resultados de uma análise de regressão podem ser bastante sensíveis a *outliers* (valores discrepantes), muitas vezes necessitando de sua remoção. Portanto, removemos esse animal do nosso dataset e realizamos testes com as CNNs pré-treinadas para avaliar o desempenho na extração de atributos para a regressão de peso em pequenos ruminantes. Os modelos *VGG16*, *VGG19* e *InceptionResNetV2* apresentaram resultados inferiores em comparação com a *Xception* e a *EfficientNetB1*. Assim, a Tabela 2 apresenta apenas os resultados dessas duas últimas arquiteturas.

**Tabela 2. Resultados dos Testes de Regressão.**

Regressor	MSE	RMSE	MAE	MRAE	MAPE	R <sup>2</sup>	r
Xception							
Linear Regression	21.72 ± 12.62	4.51 ± 1.19	3.13 ± 0.41	<b>16.00 ± 1.00</b>	<b>15.69 ± 0.75</b>	0.77 ± 0.15	0.89 ± 0.07
Random Forest	29.85 ± 4.12	5.45 ± 0.37	4.00 ± 0.22	25.00 ± 4.00	25.43 ± 3.75	0.69 ± 0.04	0.87 ± 0.04
SVR	100.28 ± 12.28	10.00 ± 0.61	7.23 ± 0.38	53.00 ± 11.00	52.78 ± 10.99	-0.04 ± 0.12	0.27 ± 0.08
Lasso	26.57 ± 7.13	5.11 ± 0.65	3.56 ± 0.45	19.00 ± 3.00	19.46 ± 3.08	0.72 ± 0.09	0.86 ± 0.04
ElasticNet	25.53 ± 8.24	4.99 ± 0.76	3.48 ± 0.40	19.00 ± 3.00	18.62 ± 2.52	0.73 ± 0.11	0.87 ± 0.04
Ridge	21.71 ± 12.61	<b>4.50 ± 1.19</b>	3.13 ± 0.41	16.00 ± 1.00	<b>15.69 ± 0.75</b>	0.77 ± 0.15	0.89 ± 0.07
Gradient Boosting	<b>21.72 ± 7.32</b>	4.60 ± 0.74	<b>3.08 ± 0.41</b>	19.00 ± 4.00	19.25 ± 3.64	<b>0.77 ± 0.07</b>	<b>0.89 ± 0.04</b>
KNeighbors	48.79 ± 7.60	6.96 ± 0.54	4.50 ± 0.43	26.00 ± 4.00	26.23 ± 2.94	0.49 ± 0.08	0.72 ± 0.04
EfficientNetB1							
Linear Regression	22.89 ± 8.95	4.69 ± 0.93	3.33 ± 0.66	18.00 ± 6.00	18.46 ± 6.32	0.76 ± 0.09	0.88 ± 0.04
Random Forest	<b>15.58 ± 4.29</b>	<b>3.91 ± 0.56</b>	2.66 ± 0.42	14.00 ± 4.00	13.98 ± 3.62	<b>0.84 ± 0.03</b>	<b>0.92 ± 0.01</b>
SVR	79.15 ± 9.10	8.88 ± 0.51	6.60 ± 0.42	45.00 ± 10.00	44.99 ± 9.60	0.18 ± 0.08	0.59 ± 0.08
Lasso	33.33 ± 11.28	5.69 ± 0.95	4.27 ± 0.55	25.00 ± 9.00	24.80 ± 8.51	0.65 ± 0.12	0.82 ± 0.07
ElasticNet	31.05 ± 9.33	5.51 ± 0.84	4.05 ± 0.57	23.00 ± 8.00	23.15 ± 7.57	0.68 ± 0.10	0.83 ± 0.05
Ridge	22.90 ± 8.95	4.69 ± 0.93	3.33 ± 0.66	18.00 ± 6.00	18.47 ± 6.33	0.76 ± 0.09	0.88 ± 0.04
Gradient Boosting	18.02 ± 4.96	4.20 ± 0.61	<b>2.65 ± 0.38</b>	<b>13.00 ± 3.00</b>	<b>12.86 ± 3.02</b>	0.81 ± 0.05	0.91 ± 0.02
KNeighbors	30.53 ± 11.54	5.43 ± 1.04	3.77 ± 0.90	20.00 ± 7.00	20.41 ± 7.42	0.68 ± 0.12	0.83 ± 0.08

A análise dos resultados revela que os regressores *Linear Regression* e *Ridge* se adaptaram melhor aos atributos extraídos pelas redes *VGG16* e *VGG19*, enquanto os modelos *Gradient Boosting* e *Random Forest* apresentaram melhor desempenho com atributos extraídos por redes mais complexas como *Xception* e *EfficientNetB1*. Esse comportamento pode ser atribuído à natureza dos atributos gerados por essas CNNs: *VGG16* e *VGG19* produzem atributos mais simples e lineares, adequados para regressores lineares como *Linear Regression* e *Ridge*. Em contraste, as redes *Xception* e *EfficientNetB1*, capturam características mais complexas e não-lineares, que são melhor interpretadas por modelos *ensemble* como *Gradient Boosting* e *Random Forest*, capazes de manejar interações complexas e relações não-lineares. Assim, a escolha do regressor deve considerar a complexidade dos atributos extraídos, utilizando métodos lineares para atributos simples e métodos mais sofisticados para atributos complexos como os deste trabalho.

Os melhores desempenhos foram obtidos com os regressores Gradient Boosting e Random Forest utilizando atributos extraídos pela EfficientNetB1. Embora o Random Forest tenha apresentado um coeficiente de determinação ( $R^2$ ) ligeiramente superior, optamos por destacar o Gradient Boosting por apresentar os menores valores de MAE ( $2,65 \pm 0,38$  kg) e MAPE ( $12,86 \pm 3,02$ ), métricas que avaliam diretamente o erro absoluto e

percentual médio nas predições, sendo mais representativas para a análise prática da estimativa de peso. O modelo Gradient Boosting, com a EfficientNetB1, obteve também um RMSE de  $4,20 \pm 0,61$  kg e um  $R^2$  de  $0,81 \pm 0,05$ . O MAE indica que os erros médios de predição são de 2,65 kg, o que representa uma margem aceitável considerando a faixa de peso dos animais (6,8 kg a 44,2 kg). Esses resultados demonstram que o modelo é eficaz na predição de pesos em condições realistas. A Tabela 3 apresenta uma comparação com os principais trabalhos presentes na literatura. Cabe destacar que cada trabalho utilizou conjuntos de dados distintos, o que impede uma comparação direta entre os resultados.

**Tabela 3. Comparação com Trabalhos Relacionados.**

Trabalho	MAE	MRAE	RMSE	MAPE	$R^2$
[Samperio et al. 2021]	—	<b>6.00</b>	—	—	<b>86.00</b>
[Sant’Ana et al. 2021]	$3.09 \pm 1.52$	—	<b><math>3.48 \pm 1.67</math></b>	<b><math>8.78 \pm 4.56</math></b>	$68.00 \pm 9.00$
[He et al. 2023]	6.73	—	8.86	14.60	—
[Menesatti et al. 2014]	$3.54 \pm 3.40$	—	—	—	—
EfficientNetB1 + Random Forest	$2.66 \pm 0.42$	$14.00 \pm 4.00$	$3.91 \pm 0.56$	$13.98 \pm 3.62$	$84.00 \pm 3.00$
EfficientNetB1 + Gradient Boosting	<b><math>2.65 \pm 0.38</math></b>	$13.00 \pm 3.00$	$4.20 \pm 0.61$	$12.86 \pm 3.02$	$81.00 \pm 5.00$

A comparação com trabalhos relacionados revela que, este trabalho incorpora técnicas de Inteligência Artificial para alcançar resultados competitivos na tarefa de regressão. Em particular, o modelo baseado na combinação da EfficientNetB1 com o Gradient Boosting obteve um MAE de  $2,65 \pm 0,38$ , superando o melhor resultado reportado na literatura analisada ( $3,09 \pm 1,52$ ) por [Sant’Ana et al. 2021].

Observa-se ainda que alguns trabalhos anteriores utilizam um número limitado de métricas e imagens, dificultando comparações mais abrangentes. Nossa estudo se diferencia ao apresentar uma avaliação mais completa, com múltiplas métricas e análise detalhada do comportamento dos modelos. Dessa forma, o sistema desenvolvido combina inovação tecnológica na construção da plataforma com avanços relevantes na aplicação de técnicas de deep learning e regressão, alinhando-se às temáticas centrais da Inteligência Artificial aplicada à pecuária de precisão.

## 5. Conclusão e Trabalhos Futuros

O sistema Web demonstrou ser uma ferramenta eficaz para a estimativa automática e precisa do peso corporal de ovinos e caprinos. Utilizando técnicas avançadas de segmentação de imagens com *U-net* e modelos de regressão baseados em CNNs pré-treinadas, o sistema obteve resultados promissores, com performance competitiva em diversas métricas de avaliação. Esta abordagem permitiu uma extração eficiente de características e predições acuradas do peso dos animais.

A plataforma Web integra um modelo preditivo robusto com uma interface acessível, que disponibiliza funcionalidades como visualização de métricas e gerenciamento de imagens por animal, oferecendo suporte à tomada de decisão no manejo dos rebanhos. Dessa forma, atende a diferentes perfis de usuários, desde pequenos produtores até grandes empresas do setor. Para aprimorar o sistema, futuras pesquisas podem focar em expandir a base de dados, explorar novas arquiteturas de redes neurais profundas e

integrar diferentes modalidades de dados. Também é importante validar o sistema em condições reais de campo e desenvolver aplicativos móveis para facilitar o acesso direto no campo. Outras áreas a serem exploradas incluem a viabilidade econômica e o impacto ambiental do sistema, bem como a criação de um modelo de negócios sustentável.

## Referências

- Abadi, M. et al. (2016). Tensorflow: A system for large-scale machine learning. In *OSDI*.
- ABIEC (2022). Beef report – brazilian livestock profile. <http://abiec.com.br/publicacoes/beef-report-2022>.
- Allen, T. et al. (2024). Django rest framework simple jwt. <https://django-rest-framework-simplejwt.readthedocs.io/>.
- Araújo, R. L., Araújo, F. H. d., and Silva, R. R. e. (2022). Automatic segmentation of melanoma skin cancer using transfer learning and fine-tuning. *Multimedia Systems*, 28(4):1239–1250.
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *arXiv preprint arXiv:1610.02357*.
- Chollet, F. et al. (2015). Keras. <https://github.com/keras-team/keras>.
- Christie, T. and Gibson, C. (2024). Django rest framework. <https://www.djangoproject-rest-framework.org/>.
- de LUCENA, C. and GUIMARAES, V. (2018). Boletim do centro de inteligência e mercado de caprinos e ovinos.
- Dickinson, R., Morton, J., Beggs, D., Anderson, G., Pyman, M., Mansell, P., and Blackwood, C. (2013). An automated walk-over weighing system as a tool for measuring liveweight change in lactating dairy cows. *Journal of dairy science*, 96(7):4477–4486.
- DROVERS (2020). World cattle inventory: Ranking of countries (fao).
- He, C., Qiao, Y., Mao, R., Li, M., and Wang, M. (2023). Enhanced litehrnet based sheep weight estimation using rgb-d images. *Computers and Electronics in Agriculture*, 206:107667.
- Iakubovskii, P. (2019). Segmentation models. [https://github.com/qubvel/segmentation\\_models](https://github.com/qubvel/segmentation_models).
- Klivans, A., Kothari, P. K., and Meka, R. (2018). Efficient algorithms for outlier-robust regression. In *Conference On Learning Theory*, pages 1420–1430. PMLR.
- Kreibich, J. A. (2010). *Using SQLite*. O'Reilly Media, Inc.
- Liu, D., He, D., and Norton, T. (2020). Automatic estimation of dairy cattle body condition score from depth image using ensemble model. *biosystems engineering*, 194:16–27.
- Martins, B., Mendes, A., Silva, L., Moreira, T., Costa, J., Rotta, P., Chizzotti, M., and Marcondes, M. (2020). Estimating body weight, body condition score, and type traits in dairy cows using three dimensional cameras and manual body measurements. *Livestock science*, 236:104054.

- Menesatti, P., Costa, C., Antonucci, F., Steri, R., Pallottino, F., and Catillo, G. (2014). A low-cost stereovision system to estimate size and weight of live sheep. *Computers and Electronics in Agriculture*, 103:33–38.
- NÓBREGA, A. and VERGNE, M. (2018). Novo censo agropecuário mostra crescimento de efetivo de caprinos e ovinos no nordeste. *Embrapa Ovinos e Caprinos*.
- of Geography, I. B. I. and Statistics (2018). Municipal livestock production. 2018.
- Oliveira, T. M., Sarmento, J. L., Figueiredo Filho, L. A., and Silva, R. R. (2021). Segmentação de imagens de ultrassonografia da carcaça em pequenos ruminantes utilizando deep learning. In *Anais do XIV Encontro Unificado de Computação do Piauí e XI Simpósio de Sistemas de Informação*, pages 144–151. SBC.
- Pedregosa, F. et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.
- 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.
- Sabbioni, A., Beretti, V., Superchi, P., and Ablondi, M. (2020). Body weight estimation from body measures in cornigliese sheep breed. *Italian Journal of Animal Science*, 19(1):25–30.
- Samperio, E., Lidón, I., Rebollar, R., Castejón-Limas, M., and Álvarez-Aparicio, C. (2021). Lambs' live weight estimation using 3d images. *Animal*, 15(5):100212.
- Sant'Ana, D. A., Pache, M. C. B., Martins, J., Soares, W. P., de Melo, S. L. N., Garcia, V., de Moares Weber, V. A., da Silva Heimbach, N., Mateus, R. G., and Pistori, H. (2021). Weighing live sheep using computer vision techniques and regression machine learning. *Machine Learning with Applications*, 5:100076.
- Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Song, X., Bokkers, E., Van der Tol, P., Koerkamp, P. G., and Van Mourik, S. (2018). Automated body weight prediction of dairy cows using 3-dimensional vision. *Journal of Dairy Science*, 101(5):4448–4459.
- Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*.
- Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*.
- Vincent, W. S. (2018). *Django for Beginners*. Independently published.
- Zhang, A. L., Wu, B. P., Wuyun, C. T., Jiang, D. X., Xuan, E. C., and Ma, F. Y. (2018). Algorithm of sheep body dimension measurement and its applications based on image analysis. *Computers and Electronics in Agriculture*, 153:33–45.