

Desenvolvimento de um Sistema para Análise de Viabilidade de Sêmen Ovino Utilizando Yolov8

Vanessa L. Nascimento¹, Robson G. F. Feitosa¹, Gabriela L. Lima¹, Harley M. Mello¹, Guilherme A. R. M. Esmeraldo¹, Raul S. Maciel²

¹ Instituto Federal de Educação, Ciência e Tecnologia do Ceará – Campus Crato (IFCE)
Rodovia CE 292, KM 15, Gisélia Pinheiro - CEP 63115-500 - Crato - CE – Brazil.

² Universidade Federal do Cariri – Campus Juazeiro do Norte (UFCA)
AV. Ten. Raimundo Rocha 1639, Cidade Universitária - CEP 63048-080 - Juazeiro do Norte - CE - Brazil.

{vanessa.lima06@aluno.ifce.edu.br, robsonfeitosa@ifce.edu.br,
gabriela.lima@ifce.edu.br, harley.mello@ifce.edu.br,
guilhermealvaro@ifce.edu.br, raul.maciel@aluno.ufca.edu.br}

Abstract. *In recent years, sheep farming has experienced significant growth, which has motivated the development of technologies to improve reproductive efficiency. This work proposed the development of a mobile application for the viability analysis of ovine semen using YOLOv8. The process included: the creation of a database with 197 images; training the model in three distinct stages; and the development of a Web server using Flask, integrated with an Android application. The best mAP50 performance achieved was 67.4%, with a precision of 70.1% and recall of 62.4%. The application presents itself as a practical, low-cost, and accessible alternative for students and researchers in Animal Science and Veterinary Medicine, using artificial intelligence in reproductive analysis.*

Resumo. *Nos últimos anos, a criação de ovinos tem experimentado um crescimento significativo, o que motivou o desenvolvimento de tecnologias para melhorar a eficiência reprodutiva. Este trabalho propôs o desenvolvimento de um aplicativo móvel para análise da viabilidade de sêmen ovino utilizando YOLOv8. O processo incluiu: a criação de uma base de dados com 197 imagens; o treinamento do modelo em três etapas distintas; e, o desenvolvimento de um servidor Web com Flask, integrado a um aplicativo Android. O melhor desempenho mAP50 alcançado foi de 67,4%, com precisão de 70,1% e recall de 62,4%. O aplicativo se apresenta como uma alternativa prática, de baixo custo e acessível para estudantes e pesquisadores de Zootecnia e Veterinária, para análise reprodutiva por meio do uso de inteligência artificial.*

1. Introdução

Segundo Pereira *et al.* (2017), nos últimos anos, a criação de ovinos tem experimentado um crescimento significativo, impulsionando a busca por tecnologias que melhorem a eficiência reprodutiva e acelere os processos de melhoramento genético. Embora os métodos mais sofisticados de análise de sêmen ovino sejam limitados a centros de pesquisa, os progressos nessa área contribuem para aprimorar a precisão e a consistência dos resultados [Bergstein, Weiss e Bicudo 2014].

Segundo Snoeck *et al.* (2020), os sistemas de análise computadorizada de sêmen (CASA) são ferramentas avançadas e padronizadas, oferecendo maior confiabilidade nos estudos de reprodução animal em comparação aos métodos tradicionais, no entanto

exigem investimentos significativos e procedimentos operacionais rigorosos para garantir precisão. O CASA possui desvantagens, como o alto custo, a necessidade de validação para diferentes espécies e a padronização do processo, conforme Davis e Katz (1993) e Verstegen et al. (2002). Além disso, enfrenta limitações, como a dificuldade em distinguir espermatozoides de outros componentes seminais e a falta de mobilidade para uso em campo, o que tem levado à exploração de metodologias alternativas, como o uso de inteligência artificial, para superar esses desafios [Panner Selvam, et al., 2024].

Embora a maioria dos laboratórios utilizem sistemas comerciais, alternativas gratuitas e de código aberto, como o OpenCASA¹, têm surgido para reduzir custos em grupos de pesquisa. O OpenCASA é um software baseado no sistema CASA, permitindo o estudo de três parâmetros clássicos de qualidade do sêmen: motilidade, morfometria e integridade da membrana (viabilidade). No entanto, Alquézar-Baeta *et al.* (2019) aponta que o sistema foi desenvolvido e testado apenas no Windows 7 (64 bits), utilizando o ImageJ v1.49q e Java JRE 1.8.0_101 (64 bits), o que limita o uso em outras plataformas. A falta de testes em sistemas Android, iOS, Linux e MacOS pode comprometer sua mobilidade e compatibilidade, dificultando a adoção em laboratórios com diferentes sistemas operacionais. Diante do exposto, o presente trabalho propõe o desenvolvimento de um aplicativo que utiliza o modelo YOLOv8 para analisar a viabilidade de sêmen ovino. Para isso, o trabalho foi organizado da seguinte forma: na Seção 2, são discutidos os trabalhos relacionados ao tema de análise de espermatozoides; na Seção 3, são listados os materiais, métodos e processos adotados para o desenvolvimento do trabalho; na Seção 4, são apresentados os detalhes de concepção e desenvolvimento do sistema; na Seção 5, são discutidos os principais resultados alcançados; por fim, na Seção 6, são apresentadas as considerações finais.

2. Trabalhos Relacionados

Azevedo *et al.* (2024) ressaltam que, com o aumento da acessibilidade ao CASA, é fundamental expandir seu uso para além da análise da motilidade, incorporando também avaliações citométricas e morfológicas. O autor observa que, embora as análises CASA forneçam uma avaliação objetiva da motilidade e morfologia do espermatozoide, elas não conseguem estimar a qualidade subjetiva ou o potencial de fertilização da amostra.

Segundo Yüzkat *et al.* (2024), diferentes estudos sobre detecção de espermatozoides utilizaram abordagens diversas e apresentaram resultados variados, conforme ilustrado na Tabela 1.

Tabela 1. Ferramentas utilizadas e resultados de estudos sobre detecção de espermatozoides, baseado em Yüzkat et al. (2024).

Fonte	Espécie	Modelo	Resultado	Base
[Hidayatullah 2021]	Touro	YOLOv3 e YOLOv4	mAP 94,11; F1 0,93; 2,18x mais rápido em teste, 2,9x em treino; menor uso de memória e tamanho de arquivo de pesos 20x menor que o YOLOv4.	Público. Disponível em: Github
[Yüzkat 2024]	Humano	Faster R-CNN e YOLOv3	YOLOv3 apresentou melhor desempenho em precisão (mAP de 84,5% a 96%) e tempo de treino em relação ao Faster	Base de dados privada.

¹ <https://github.com/calquezar/OpenCASA>

			R-CNN, especialmente em vídeos individuais.	
[Valiuskaite 2024]	Humano	R-CNN + Algoritmo de Rastreamento CCTA	Precisão de 91,77% na segmentação da cabeça do espermatozoide; não comparou com YOLO.	Público. Disponível em: VISEM
[Zou <i>et al.</i>]	Não especificado	Tiny Object Detection-CNN (TOD-CNN)	85% mAP em 111 vídeos de espermatozoides.	Base de dados privada.
[Rahimzadeh <i>et al.</i>]	Não especificado	RetinaNet + Filtros DCF	Precisão média de 96%; desempenho de detecção comparável aos estudos atuais.	Base de dados privada.
[Chen <i>et al.</i>]	Humano	YOLO, SSD, RetinaNet, Faster R-CNN	YOLO alcançou o maior mAP de 40%, embora o valor seja baixo em comparação com outros estudos	Pública, disponível em: Github
[Aggarwal <i>et al.</i>]	Humanos (pacientes com câncer)	YOLOv3 e Faster R-CNN	mAP de 0,37 com YOLOv3 e 0,56 com Faster R-CNN; tempo de processamento maior.	Base de dados privada.
[Mifune <i>et al.</i>]	Humano	YOLOv5	81% mAP no conjunto de dados de espermatozoides micro-TESE; melhor em termos de sucesso e tempo.	Base de dados privada.
Proposta deste trabalho	Ovino	YOLOv8	Melhor mAP50 de 67,4%; mAP50-95 de 32,7%; Precisão de 70,1%; Recall de 62,4%.	Pública, disponível em: Driver

Segundo Yüzkat *et al.* (2024), a limitação dos estudos sobre a detecção de espermatozoides e análise de movimento deve-se à escassez de conjuntos de dados específicos de espermatozoides na literatura.

3. Materiais e Métodos

O desenvolvimento deste trabalho foi estruturado nas seguintes etapas: (1) definição da motivação e dos objetivos do estudo; (2) análise da literatura para embasamento teórico; (3) descrição dos materiais e métodos utilizados; (4) análise de requisitos e desenvolvimento do sistema; e, (5) apresentação e discussão dos resultados obtidos.

O treinamento do modelo de visão computacional adotado neste trabalho (YOLOv8) foi realizado em um desktop com processador AMD Ryzen 5 4600G (6 núcleos, 12 threads), 16GB de RAM DDR4 e gráficos integrados Radeon Vega. Foram realizadas 3 etapas de treinamento com diferentes critérios detalhados na Tabela 2, onde no primeiro treinamento as imagens foram utilizadas no formato original e com resolução baixa, no segundo foram utilizadas as mesmas imagens com qualidade maior, e no terceiro treinamento as imagens passaram por um processo de *Data Augmentation*. O processo de treinamento do modelo demandou, aproximadamente, 82 horas, somando o tempo dos três treinamentos. Todos os treinamentos foram realizados com o parâmetro *react* valor *True* e com 1000 épocas.

Tabela 2. Configuração de treinamento

Treino	Épocas concluídas	Melhor época	Imagens treino	Tamanho	Batch
1º	592	492	130	640x640	16
2º	637	537	130	1280x1280	8
3º	228	178	530	1280x1280	8

4. Análise de requisitos e desenvolvimento

Entre os requisitos funcionais identificados, destaca-se a necessidade de identificar e contar células coradas e não coradas, com uma contagem mínima de 200 células por análise. Além disso, o sistema deve permitir a captura de imagens pela câmera e a seleção de até cinco imagens da galeria.

4.1 Arquitetura do sistema

Para a solução aqui proposta, adotou-se uma arquitetura de software em camadas, caracterizada pela separação clara de responsabilidades em diferentes níveis. Essa abordagem foi escolhida visando uma melhor organização do código, facilitando a manutenção, escalabilidade e reutilização dos componentes. Cada camada desempenha um papel bem definido, garantindo uma estrutura modular e de fácil compreensão.

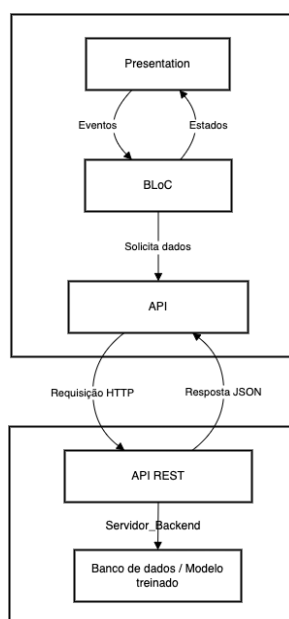


Figura 1. Diagrama de arquitetura de camadas

4.2 Construção da base de dados

A revisão da literatura mostrou que a maioria dos estudos cria suas próprias bases de dados devido à limitação de conjuntos públicos, especialmente na produção animal. Neste estudo, a base de dados foi composta por 197 imagens capturadas por smartphones acoplados a um microscópio (400×) durante aulas no IFCE campus Crato. Destas, 80% foram usadas para treinamento e 20% para validação e testes.

4.3 Processamento das imagens

O processamento das imagens para o terceiro treinamento foi realizado utilizando a técnica de *Data Augmentation*, onde, segundo Salvador *et al.* (2024), a técnica de aumento de dados permite criar variações das imagens originais, enriquecendo a base e promovendo maior robustez e generalização dos modelos. As imagens foram editadas para formato quadrado, focando nas regiões relevantes. No total, 663 imagens foram processadas, sendo 530 para treinamento e 133 para validação e testes.

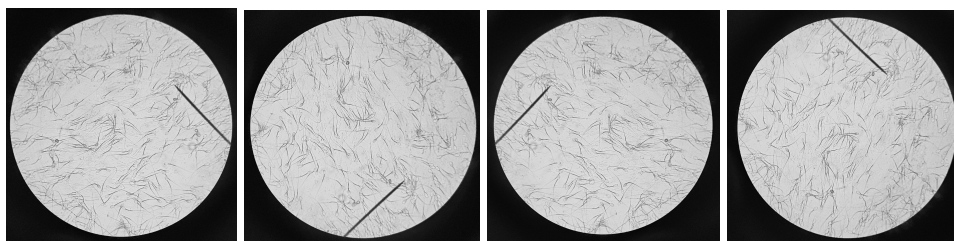


Figura 2. Número de dados aumentados com uso de Data Augmentation

4.4 Rotulação

Uma das principais etapas deste trabalho consiste na rotulação manual das imagens de treinamento e validação, na qual utilizou-se a ferramenta Label Studio. Espermatozoides mortos foram classificados como "corados", enquanto os vivos foram considerados "não-corados". A figura 3 mostra as características dessas células.

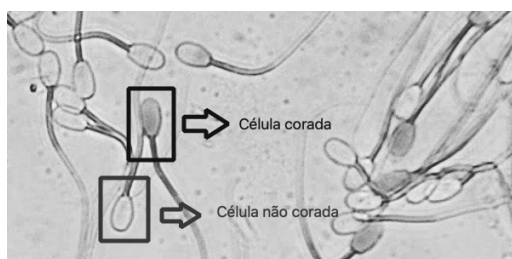


Figura 3. Definição de célula corada e não corada.

4.5 Validação do modelo

Para a validação do modelo, foram separadas 133 imagens. Esse processo é fundamental para garantir que o modelo não tenha apenas memorizado os dados de treinamento, mas também aprendido a identificar células de maneira generalizada, possibilitando um desempenho confiável em dados novos e desconhecidos. Os resultados gerais da validação dos três treinamentos realizados estão apresentados na Tabela 3.

Tabela 3. Desempenho Geral dos Modelos nos Três Treinamentos Durante a Validação

Treinamento	mAP50 (%)	mAP50-95 (%)	Precisão (%)	Recall (%)
1º	31,1	10,2	67,5	31,4
2º	35,2	12,4	72,4	33,8
3º	67,4	32,7	70,1	62,4

4.6 Desenvolvimento do Sistema

O aplicativo para sistema Android foi desenvolvido utilizando o framework Flutter com a linguagem Dart. Flutter é um kit de ferramentas de UI multiplataforma que foi projetado para permitir a reutilização de código em sistemas operacionais como iOS e Android, ao mesmo tempo em que permite que aplicativos interajam diretamente com serviços de plataforma subjacentes [Velichko 2024]. Inicialmente, o aplicativo foi desenvolvido para a plataforma Android, mas a escolha do Flutter foi estratégica, pois este framework facilita a expansão futura para o iOS, devido ao seu suporte nativo a múltiplas plataformas.

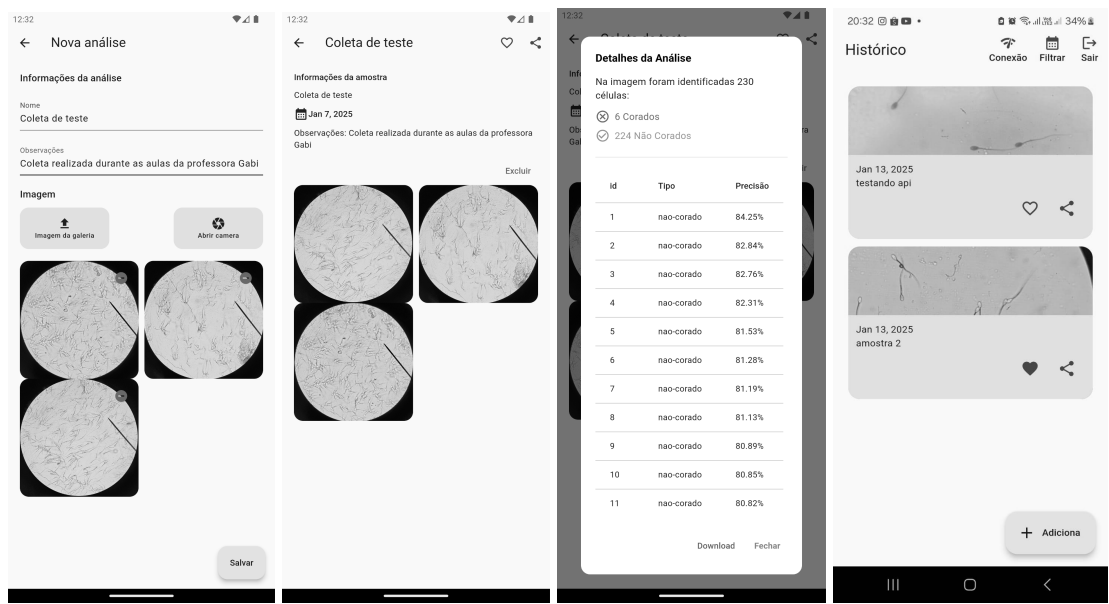


Figura 4. Principais funcionalidades da aplicação

A linguagem Python possui grande versatilidade, podendo atuar bem na área comercial, ou em áreas mais específicas como por exemplo: no desenvolvimento científico, geoprocessamento e em aplicações mobile, tanto isoladamente ou integrada a outras ferramentas [Silva 2019]. No contexto deste projeto, Python foi utilizado para desenvolver o servidor Web por meio da biblioteca Flask. O Flask é um framework de aplicativo Web WSGI leve. Ele foi projetado para tornar o início rápido e fácil, com a capacidade de escalar para aplicativos complexos [Flask 2010].

5. Resultados e Discussões

A Tabela 4 resume as melhores épocas de treinamento dos modelos. O terceiro treinamento obteve o melhor desempenho geral na época 178 (mAP50: 65,375%, recall: 62,446%), superando os anteriores. O primeiro teve desempenho inferior (mAP50: 36,454%, recall: 29,7417%) na época 492, enquanto o segundo, na época 537, apresentou mAP50 de 52,595% e recall de 52,595%. O terceiro modelo destacou-se pela rápida convergência e melhor generalização.

Tabela 4. Resumo das Melhores Épocas Identificadas Durante o Treinamento dos Modelos

Treinamento	Melhor Época	mAP50 (%)	mAP50-95 (%)	Precisão (%)	Recall (%)
1º	492	36,454	13,9288	65,214	29,7417
2º	537	52,595	19,975	59,407	52,595
3º	178	65,375	32,664	69,993	62,446

A Tabela 5 apresenta o desempenho dos modelos treinados na validação. O primeiro teve baixa generalização (mAP50: 31,1%, recall: 31,4%). O segundo mostrou melhorias na precisão (72,4%), mas com recall ainda baixo (33,8%). O terceiro modelo obteve os melhores resultados (mAP50: 67,4%, recall: 62,4%), destacando-se pelo equilíbrio entre precisão e recall.

Tabela 5. Desempenho Geral dos Modelos Durante a Validação

Modelo	mAP50 (%)	mAP50-95 (%)	Precisão (%)	Recall (%)
1º	31,1	10,2	67,5	31,4
2º	35,2	12,4	72,4	33,8
3º	67,4	32,7	70,1	62,4

6. Conclusão

Este trabalho apresentou o desenvolvimento de um sistema de baixo custo para a análise de sêmen ovino, utilizando o modelo YOLOv8. Os resultados do treinamento evidenciaram a eficácia do *Data Augmentation* na melhoria da precisão, recall e mAP. O terceiro modelo, com uma base de dados ampliada, obteve desempenho superior, com mAP50 de 67,4% e recall de 62,4%. No entanto, algumas limitações foram identificadas, como a necessidade de maior diversificação da base de dados. Como sugestões de trabalhos futuros, o desenvolvimento de funcionalidades para análise de motilidade através de vídeos, utilizando técnicas avançadas de processamento de imagem, pode agregar mais benefícios aos usuários do sistema.

Os resultados mostraram que o uso de *Data Augmentation* no treinamento do modelo de detecção de espermatozoides trouxe melhorias significativas em precisão, recall e mAP, aprimorando a avaliação da viabilidade espermática com corantes vitais. O terceiro modelo, com uma base de dados ampliada, obteve o melhor desempenho, destacando que a combinação de inteligência artificial com metodologias estruturadas pode tornar a análise de sêmen ovino mais eficiente e acessível. Esses resultados ressaltam o potencial das redes neurais na biotecnologia reprodutiva e apontam para possíveis melhorias futuras no sistema.

Referências

Alquézar-Baeta, C., Gimeno-Martos, S., Miguel-Jiménez, S., Santolaria, P., Yániz, J., et al. (2019) “OpenCASA: A new open-source and scalable tool for sperm quality analysis”, *PLOS Computational Biology*, v. 15, n. 1, p. e1006691. Disponível em: <https://doi.org/10.1371/journal.pcbi.1006691>.

- Azevedo, H. C., Blackburn, H. D., Lozada-Soto, E. A., Spiller, S. F., Purdy, P. H. (2024) “Enhancing evaluation of bull fertility through multivariate analysis of sperm”, *Journal of Dairy Science*, ISSN 0022-0302. Disponível em: <https://doi.org/10.3168/jds.2024-25163>.
- Bergstein, T. G., Weiss, R. R., Bicudo, S. D. (2014) “Técnicas de análise de sêmen”, *Revista Brasileira de Reprodução Animal*, Belo Horizonte, v. 38, n. 4, p. 189-194, out./dez. Disponível em: www.cbra.org.br.
- Flask (2010) “Documentação oficial do Flask”. Disponível em: <https://flask.palletsprojects.com/en/stable/>.
- Hidayatullah, P., Wang, X., Yamasaki, T., Mengko, T. L. E. R., Munir, R., Barlian, A., Sukmawati, E., Suprptono, S. (2021) “DeepSperm: A robust and real-time bull sperm-cell detection in densely populated semen videos”, *Computer Methods and Programs in Biomedicine*, v. 209, p. 106302. Disponível em: <https://doi.org/10.1016/j.cmpb.2021.106302>.
- Pereira, L. M. C., et al. (2017) “Inseminação artificial em ovinos e caprinos”, *Anais da Semana do Curso de Zootecnia-SEZUS*, Brasil, v. 11, p. 1.
- Salvador, P. H., Pinheiro, A. L. A., Feitosa, F. C. D. C., Feitosa, R. G. F. (2024) “Rede Neural Convolucional para a Classificação de Imagens de Pessoas Fazendo Uso de Máscaras do Tipo EPI”, *Instituto Federal de Educação, Ciência e Tecnologia do Ceará – Campus Crato*.
- Silva, R. O., Silva, I. R. S. (2019) “Linguagem de Programação Python”, *Tecnologias Em Projeção*, [S. l.], v. 10, n. 1, p. 55–71. Disponível em: <https://projecaociencia.com.br/index.php/Projecao4/article/view/1359>.
- Snoeck, P. P. N., Pessoa, T. H. O., Barros, C. H. S. C., Allaman, I. B. (2020) “Padronização do Sperm Class Analyzer® (SCA®) para avaliação de espermatozoides equinos criopreservados”, *Arquivo Brasileiro de Medicina Veterinária e Zootecnia*, 72(4), 1163–1171. Disponível em: <https://doi.org/10.1590/1678-4162-11665>.
- Velichko, I. (2024) “Flutter architectural overview”. Disponível em: <https://github.com/flutter/Website/blob/main/src/content/resources/architectural-overview.md>.
- Yüzkat, M., İlhan, H. O., Aydın, N. A. (2024) “Veriler Yapıları ve Algoritmalar”, *Muş Alparslan Üniversitesi, Yıldız Teknik Üniversitesi*, 30 ago. Yıl 2024 Cilt: 30 Sayı: 4.