

Ministério da Educação - Secretaria de Educação Profissional e Tecnológica  
Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina

Especialização em Automação Industrial

Unidade Curricular: Trabalho de conclusão de curso

### **Trabalho de Conclusão de Curso (TCC): Artigo Científico**

<b>IDENTIFICAÇÃO</b>
Ministério da Educação - Secretaria de Educação Profissional e Tecnológica Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina
Especialização em Automação Industrial
Campus: Lages
Professor orientador: Rogério da Silva
Autor: José Victor Ortiz
Título e Subtítulo: Contagem de Suínos Utilizando Visão Computacional.

## 1. INTRODUÇÃO

O avanço das tecnologias de inteligência artificial e visão computacional tem proporcionado diversas inovações no setor agroindustrial, destacando-se pela sua capacidade de otimizar processos e melhorar a eficiência operacional. Neste contexto, o presente estudo aborda o desenvolvimento de um programa em Python utilizando o modelo de detecção de imagens YOLOV8, com o objetivo específico de identificar e contar suínos em um frigorífico. A contagem precisa dos suínos em uma etapa específica do processo industrial visa fornecer dados estatísticos essenciais para o controle de qualidade e gestão produtiva (SANGOI, 2014).

Além de facilitar o controle e registro na parte do processo estudado, o programa abre precedentes para a aplicação da mesma metodologia em outros pontos da cadeia produtiva, onde seu impacto seria ainda mais expressivo. Esta abordagem pode ser estendida não só frigoríficos, como também a diferentes segmentos da indústria, possibilitando uma melhora no levantamento de dados, acompanhamento e registro de diferentes processos (CARRIJO GUIMARÃES *et al.*, 2023). Portanto, este trabalho não só contribui para a melhoria dos processos existentes, mas também estabelece uma base sólida para futuras inovações e aplicações tecnológicas no setor agroindustrial.

Com base nos trabalhos, foi possível estipular os seguintes objetivos: acuracidade superior à de 85%, precisão acima de 90% e recall de no mínimo 90%. (PANDORFI *et al.*, 2020)

## 2. DESENVOLVIMENTO

Para o desenvolvimento desse trabalho, observou-se uma sequência de atividades visando a otimização do estudo e dos resultados da pesquisa. A estrutura elaborada é apresentada na figura 1 e está disposta ao longo desse trabalho.

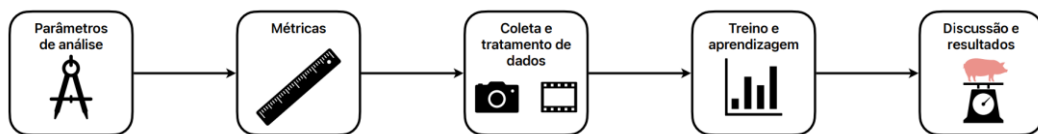


Figura 1 - Fluxograma de trabalhos (AUTOR, 2025).

### 2.1 PARÂMETROS DE ANÁLISE

Antes de apresentar o desenvolvimento do trabalho, é importante definir os conceitos que serão avaliados nos resultados desse estudo. No desenvolvimento de um novo modelo de identificação, quatro resultados distintos são esperados, e a partir da quantidade desses resultados podem ser definidas as características do modelo, se é preciso, pouco eficiente, exato, entre outros.

O conceito fica evidente na Figura 2, onde os quatro tipos de resultados esperados são apresentados.

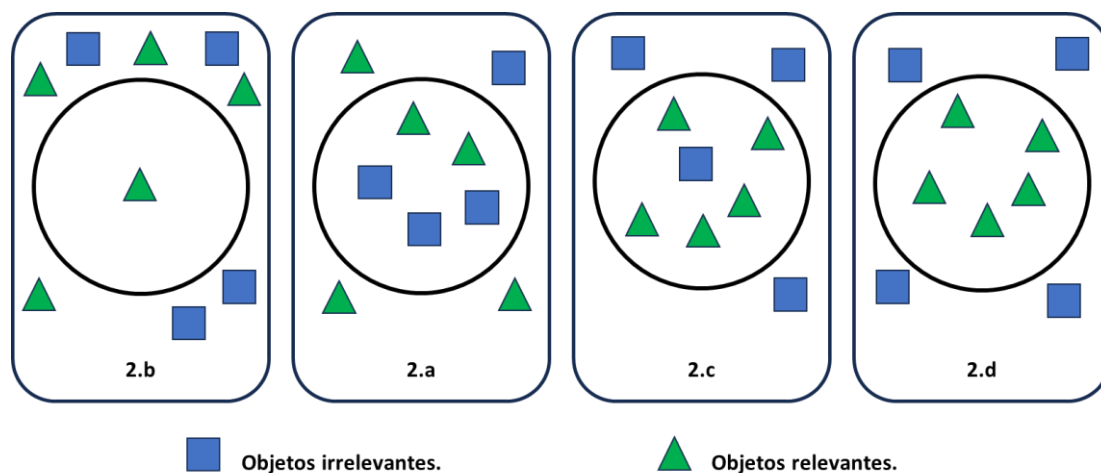


Figura 2 - Quadro de resultados possíveis (AUTOR, 2025).

No quadro 2.a da Figura 2, observa-se a identificação somente dos itens relevantes, mas não todos, sendo esse um sistema exato, porém impreciso. Já no quadro 2.b, temos um modelo não exato e abrangente no qual apenas uma parte dos itens relevantes é identificada e há itens irrelevantes que são retornados. No quadro 2.c, há um cenário mais próximo de uma etapa avançada de desenvolvimento, no qual o modelo traz todos os itens relevantes para o estudo, mas não somente eles. Por fim, no quadro 2.d, é possível identificar o cenário perfeito, no qual todos os itens relevantes são identificados e somente eles, o que nos traz um modelo exato e preciso.

Com as possibilidades de resultado claras, é possível realizar análises estruturadas sobre esses números. Sendo assim, para aferição da performance de um sistema de identificação, são geralmente utilizados os conceitos de precisão (em inglês *precision*), recall (apresentado sobre a palavra recall), acurácia e F1, todos calculados a partir das quatro diferentes possibilidades de resultados.

## 2.2 MÉTRICAS

Para melhor entendimento dos conceitos, apresenta-se abaixo uma matriz de resultados possíveis para os treinos realizados pelo modelo, sendo VP verdadeiro positivo, FN falso negativo, FP falso positivo e VN verdadeiro negativo.

		Verdadeiro	
		PORCO	NÃO PORCO
Predito	PORCO	<b>Verdadeiro Positivo - VP</b>	<b>Falso Positivo - FP</b>
	NÃO PORCO	<b>Falso Negativo - FN</b>	<b>Verdadeiro Negativo - VN</b>

Figura 3 - Resultados possíveis no modelo (AUTOR, 2025).

Ainda abaixo temos mais uma imagem que apresenta o conceito de Precisão e Recall.

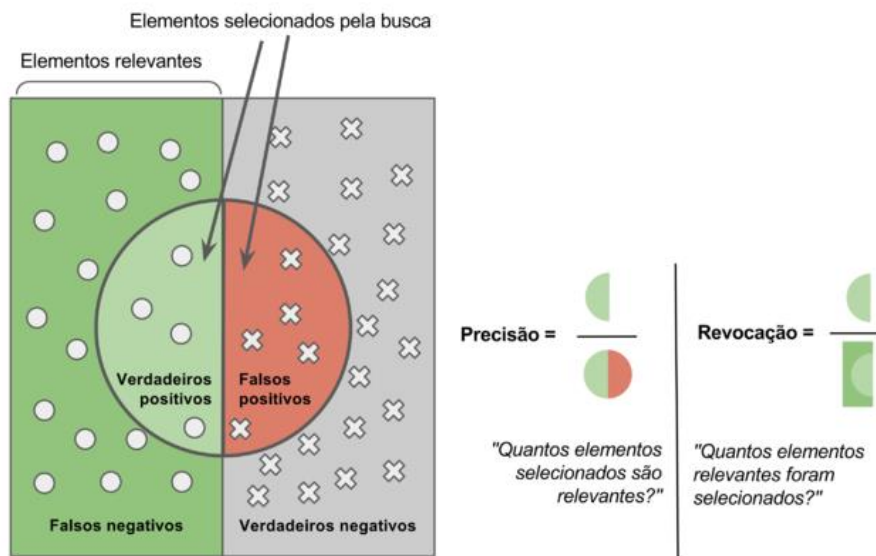


Figura 4 - Precisão e recall (BRAGA, 2017).

Idealmente, o modelo deve retornar o maior número possível de resultados verdadeiros, de modo que o sistema seja preciso e exato. Entretanto, variações são esperadas e são levadas em conta para os indicadores finais de desempenho, como consta nas fórmulas apresentadas abaixo (MARIANO, 2021).

$$Acurácia = \frac{VP + VN}{VP + FP + FN + VN} \quad [1]$$

$$Precisão = \frac{VP}{VP + FP} \quad [2]$$

$$Revocação = \frac{VP}{VP + FN} \quad [3]$$

$$F1 = \frac{2 * Precisão * Recall}{Precisão + Recall} \quad [4]$$

Sendo,

- VP Verdadeiro Positivo
- VN Verdadeiro Negativo
- FP Falso Positivo
- FN Falso Negativo

Com base no estudo de ALIF; HUSSAIN, 2024, definiu-se para esse estudo uma métrica alvo de F1 superior a 90% como resultado satisfatório para o desenvolvimento do modelo em questão (MONTEIRO *et al.*, 2016).

### 2.3 COLETA E TRATAMENTO DE DADOS

Para o desenvolvimento deste trabalho, inicialmente foram gravados vídeos no ambiente do frigorífico para capturar imagens dos suínos durante o processo produtivo em quatro etapas distintas. As quatro etapas apresentam o objeto de estudo a partir de diferentes perspectivas, com diferentes interferências e distâncias. Toda essa variabilidade permite o desenvolvimento de um modelo mais eficiente nas diferentes realidades de uso da ferramenta. Nas Figuras 5, 6, 7 e 8 constam os quatro cenários que serviram de base para o estudo.



Figura 5 - Imagem posição 1 (AUTOR, 2025).

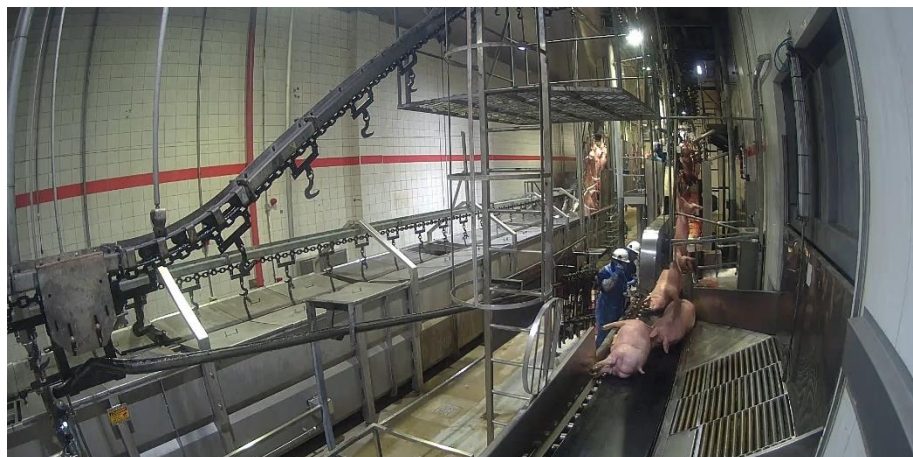


Figura 6 - Imagem posição 2 (AUTOR, 2025).



Figura 7 - Imagem posição 3 (AUTOR, 2025).



Figura 8 – Imagem posição 4 (AUTOR, 2025).

Os vídeos gravados nessas condições foram posteriormente convertidos em imagens, permitindo a anotação detalhada dos dados anteriormente registrados. Utilizando a ferramenta *Computer Vision Annotation Tool* (CVAT) e *Roboflow*, as imagens extraídas dos vídeos foram “anotadas” para identificar os suínos em cada uma das imagens. O termo “anotadas”, refere-se ao registro de coordenadas que localizam os objetos na imagem, permitindo que o programa possa utilizar essas informações em treinos, testes e validações. (ULTRALYTICS, © 2025, 2025)

Na Figura 9 um exemplo de uma das imagens em anotação.



Figura 9 - Imagem em processo de anotação (AUTOR, 2025).

As anotações realizadas incluíram a demarcação das áreas de interesse e a atribuição de “quadros” ou “formas poligonais” para cada suíno detectado. A partir disso, foi constituído um banco de dados que serviu de base para o treinamento, validação e testes do modelo. Esta divisão foi feita para que a ferramenta YOLO pudesse utilizar o material que foi coletado.

## 2.4 TREINO E APRENDIZAGEM

O modelo de detecção YOLOV8 (sigla para *You Only Look Once*), disponibilizado pela companhia *Ultralytics*, foi integrado ao código *Python* desenvolvido para esse trabalho. Esse modelo (YOLO) utiliza Redes Neurais Convolucionais para detecção de objetos e desenvolve a tarefa como um problema único de regressão, permitindo a previsão para novas detecções (ARAÚJO, 2016). A escolha desse modelo, deu-se pelo notório desempenho desse tipo de rede em tarefas relacionadas a classificação e reconhecimento de imagens (KIM, 2022).

Apesar de o modelo pré-treinado disponibilizado pela empresa ser capaz de detectar diversas classes, foi necessário desenvolver um novo modelo para otimização dos resultados específicos deste estudo, sendo identificação e contagem de suínos.

Para tanto, diferentes configurações de treinamento foram testadas, incluindo variações no número de iterações, posicionamentos de registro de imagens, versões do modelo YOLO e YOLOV8, ferramentas de anotação e imagens anotadas. Após cada rodada de treinamento, as métricas de desempenho, incluindo precisão (precision), recall e F1-score, foram calculadas e analisadas para cada configuração testada.

No início do trabalho, os resultados de detecção eram insuficientes para uma aplicação eficiente. Os testes iniciaram utilizando apenas imagens das Figuras 5 e 6, com o modelo mais compacto do YOLOV8 e com 10 iterações, resultando na ausência de detecções pelo modelo, mas com um indicativo de aprendizado. Em seguida, foram realizados testes com as imagens das Figuras 7 e 8 em 10 iterações, com um modelo pré-treinado pelo autor, retornando identificações de um número baixo de amostras no vigésimo quarto treino.

No trigésimo sétimo teste, o programa de treino fez 3 iterações e chegou aos valores de acuracidade de 20%, precisão de 25%, recall de 50% e F1 de 33%, fato que inviabilizaria a continuidade do projeto. Por outro lado, esses valores proporcionaram uma identificação de pontos de melhoria para os testes seguintes.

Na Figura 10 consta uma imagem gerada a partir de um dos primeiros testes de validação do modelo desenvolvido pelo autor.



Figura 10 - Suínos identificados após treinamento (AUTOR, 2025)

Nota-se que nos testes iniciais houve uma baixa confiabilidade do sistema desenvolvido, o que se evidencia na Figura 10 com os valores de exatidão abaixo de 50%. Com a imagem resultante do treino e testes do programa, foram feitas novas tentativas, elevando o número de iterações à medida que as configurações do desenvolvimento se tornavam mais apuradas.

Exemplo disso foi o sexagésimo quinto teste, no qual foram feitas 50 iterações e os resultados chegaram à 56% de acuracidade, 66% de precisão, 79% de recall e 72% de F1. Apesar da melhora, os números não apontaram uma eficiência significativa para justificar o uso da ferramenta em um meio industrial.

Após mais pesquisas e testes, identificou-se que a distância do ponto de registro das imagens e os objetos de estudo estavam prejudicando a identificação dos pontos anotados. Além disso, o número de obstáculos e interferências também estavam atrapalhando o desempenho do modelo. A partir disso, foram feitas novas imagens mais próximas do objeto de estudo e com menos interferências. A forma de anotação também foi alterada de quadros para polígonos, o que proporcionou uma melhora nos resultados obtidos. Com as alterações, o septuagésimo teste obteve acuracidade de 60%, precisão de 60%, recall de 100% e F1 de 75%.

Finalmente as melhores configurações foram identificadas com base nas métricas de desempenho obtidas. Após, um teste final foi realizado utilizando as configurações otimizadas determinadas nas etapas anteriores, resultando em uma acuracidade de 89%, precisão de 92%, recall de 96% e f1 de 94%.

Na Tabela 1, é possível observar as configurações utilizadas para o desenvolvimento do trabalho em sua versão com os melhores resultados.

Número de iterações.	100
Tipo de rede neural	Convolutacional
Modelo de detecção utilizado	YOLOV8
Parâmetros para fim do algoritmo.	Número de iterações.
Taxa de aprendizagem.	0,01
Número de camadas.	53

Imagens de aprendizado.	189
Imagens de validação.	55
Imagem de teste.	27

Tabela 1 – Configurações utilizadas pelo autor.

Há também algumas imagens do modelo apresentadas na Figura 11.



Figura 11 - Melhores resultados alcançados (AUTOR, 2025).

## 2.5 DISCUSSÃO E RESULTADOS

Ao longo do estudo foram identificados diversos fatores de interferência, especialmente na etapa de aprendizado e treino do modelo. Ficou evidente que as imagens utilizadas para esses fins e também a forma de anotação dessas imagens interfere consideravelmente nos resultados obtidos pelo programa.

Nesse sentido, identificou-se que a posição 3, Figura 12, foi a melhor posição, tanto para treino, quanto para testes.

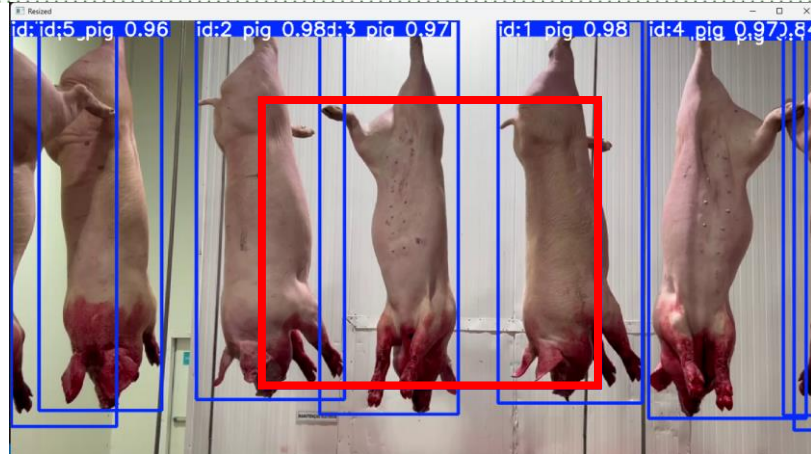


Figura 12 - Identificação e contagem de suínos de forma dinâmica (AUTOR, 2025).

Na Figura 12 é possível perceber que os níveis de exatidão são bastante elevados (como representado nos percentuais acima dos quadros na Figura 12), fato que também se comprova na execução do programa de contagem desenvolvido pelo autor.

A contagem ocorreu de forma dinâmica, através da execução do programa analisando um vídeo de forma dinâmica e instantânea. Após encerrar a operação foi possível observar que a contagem de suínos que passaram pela região retangular indicada em vermelho na Figura 12, ocorreu de maneira correta, conforme ilustrado na imagem abaixo.

```

IdentificadorREV02.py U x main.py U ! Dados_RoboFlow.yaml
Code > IdentificadorREV02.py > ...
42 # Adicionando prints para checagem
43 print("Boxes detected:", boxes)
44 print("Class IDs detected:", class_ids)
45 print("IDs detected:", ids)
46 else:
47     continue
48
49 # Conta os "pigs" dentro do quadrado delimitado
50 for box, class_id, pig_id in zip(boxes, class_ids, ids):
51     x, y, w, h = box
52     if quad_x_min < x < quad_x_max and quad_y_min < y < quad_y_max:
53         #if class_id == 0 and pig_id not in pigs_tracked: # Assegure-se de que 'pig' seja identificado como classe '0'
54             contador_pigs = pig_id + 1
55             pigs_tracked.add(pig_id)
56             print(f"Pig detected: ID {pig_id}")
57
58 # Exibe o frame anotado
59 annotated_frame = results[0].plot()

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Speed: 4.1ms preprocess, 1374.9ms inference, 0.0ms postprocess per image at shape (1, 3, 384, 640)
Boxes detected: [[ 852.9 308.18 237.03 616.23]
 [ 367.77 311.96 241.87 623.91]
 [ 562.54 320.67 230.56 641.28]
 [ 1119.9 322.04 251.76 643.93]
 [ 102.38 318.75 190.56 637.49]
 [ 1233.4 313.34 92.058 626.03]
 [ 30.241 324.16 60.482 648.33]]
Class IDs detected: [0, 0, 0, 0, 0, 0, 0]
IDs detected: [1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 12.0]
Pig detected: ID 1.0
Pig detected: ID 4.0
Pig detected: ID 6.0
Número total de pigs identificados que passaram pelo quadrado: 7.0
Número total de pigs identificados que passaram pelo quadrado: 7.0
PS C:\Z\2024\PosIFSC\TCC\Python>
Ln 50, Col 65 Spaces: 4 UTF-8 CRLF Python 3.11.8 (Microsoft Store)

```

Figura 13 - Programa de contagem de suínos (AUTOR, 2025).

Na Figura 14 são apresentados os gráficos de resultado do último treino realizado (nonagésimo nono).

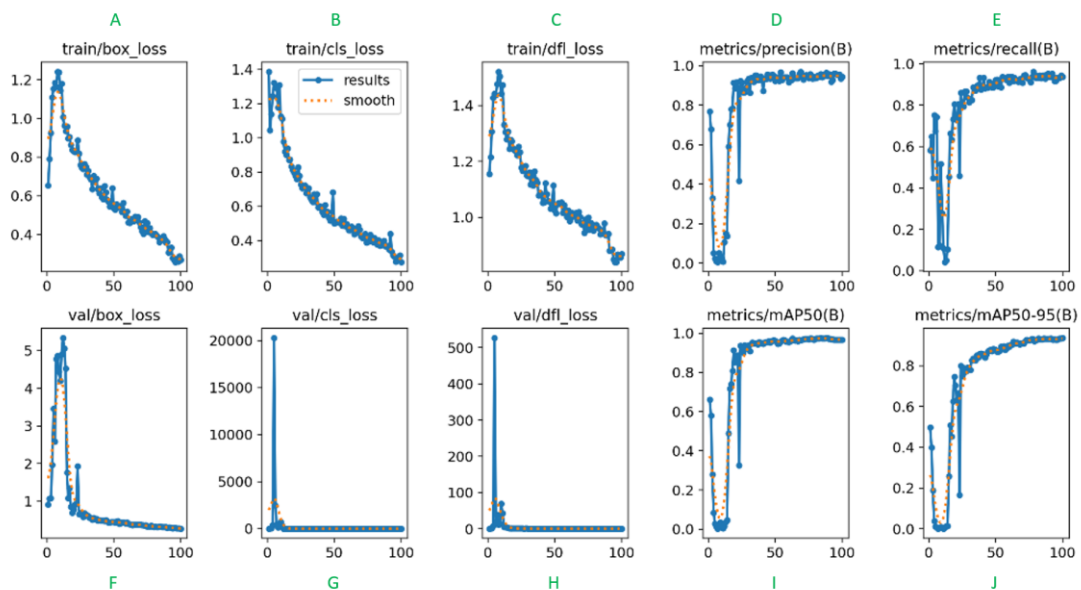


Figura 14 - Gráficos de resultado (AUTOR, 2025).

A Figura 14 acima está dividida em 10 gráficos distintos, os quais indicam o desempenho do modelo desenvolvido, sendo que no eixo “x” estão representadas as iterações do programa e no eixo “y” o parâmetro que está sendo analisado.

Nos gráficos A, B e C é possível identificar nas imagens de treino uma curva decrescente, a qual indica a regressão dos itens *box\_loss*, *cls\_loss* e *dfl\_loss* (*distribution focal loss*) que representam, o erro na localização das caixas de identificação na imagem, erro na classificação de objetos (nesse caso porcos e *background*) e erro entre objetos semelhantes respectivamente.

Nas curvas D e E, temos os gráficos de precisão e recall, indicando uma melhora em ambas as métricas, enquanto nos gráficos F, G e H temos os mesmos indicadores que nas curvas superiores, mas em relação às imagens de validação. Por sua vez, no gráfico I há uma representação do mAP50 que é a precisão média calculada com um limiar de intersecção sobre união (IoU) de 0,50. Ou seja, é uma medida da precisão do modelo considerando apenas as detecções "fáceis". Por outro lado, no gráfico J é representado o mAP50-95 que corresponde à média da precisão média calculada em diferentes limiares de IoU, variando de 0,50 a 0,95. Ou seja, dá uma visão abrangente do desempenho do modelo em diferentes níveis de dificuldade de detecção (ULTRALYTICS, 2025).

Vale ressaltar aqui que a utilizando a versão mais completa do YOLOV8, a versão *extra large* (YOLOV8x) desempenhou melhor em relação à versão *nano* (YOLOV8n), o que se observa na diferença dos indicadores abaixo. Apesar disso, o tempo de processamento na versão mais completa foi em torno de duas vezes maior no dispositivo utilizado para esse estudo.

Com YoloV8n em 100 iterações foram obtidos os resultados apresentados na Figura 15.

		Verdadeiro	
		PORCO	NÃO PORCO
Preditto	PORCO	<b>185</b>	<b>28</b>
	NÃO PORCO	<b>9</b>	<b>0</b>

acuracidade	83%
precisão	87%
recall	95%
F1	91%

Figura 15 - Matriz de confusão YoloV8n (AUTOR, 2025).

Agora, usando YoloV8X, os resultados para 100 iterações são apresentados na Figura 16.

		Verdadeiro	
		PORCO	NÃO PORCO
Preditto	PORCO	<b>186</b>	<b>16</b>
	NÃO PORCO	<b>8</b>	<b>0</b>

acuracidade	89%
precisão	92%
recall	96%
F1	94%

Figura 16 - Matriz de confusão YoloV8x (AUTOR, 2025).

Além do modelo treinado e testado pelo autor, também foi realizado um treinamento em uma nova ferramenta disponibilizada pela empresa Roboflow, na qual as imagens puderam ser anotadas e um modelo pode ser treinado na estrutura da empresa. Através dessa ferramenta, o modelo foi treinado em uma velocidade quase 150 vezes menor e os resultados também foram melhores. Abaixo as métricas disponibilizadas na plataforma, sendo mAP a média de precisão de todas as classes. Como há apenas uma, ela é igual a precisão da classe 0 (porcos).

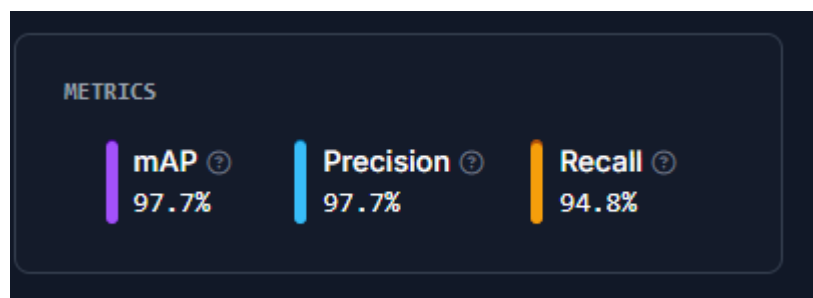


Figura 17 - Resultados modelo Roboflow (AUTOR, 2025).

### 3. CONSIDERAÇÕES FINAIS

Com o trabalho realizado, foi possível observar que a solução implementada apresenta um grande potencial de melhoria e otimização, apesar dos resultados já obtidos. A detecção, reconhecimento e contagem de suínos utilizando visão computacional demonstrou ser uma ferramenta de grande amplitude, com aplicações que podem causar um impacto significativo na indústria alimentícia, bem como em outros setores.

Levando em conta os resultados obtidos, é possível afirmar que os objetivos propostos foram alcançados, haja vista que a acuracidade foi superior à 85%, precisão maior que 90%, recall acima de 90% e F1 superior à 90%.

Para futuros trabalhos, recomenda-se considerar melhorias como a utilização de uma vista superior ou frontal para registro de imagens, a utilização de versões mais atualizadas das ferramentas utilizadas (Roboflow em especial), o aumento do número

de iterações para treinamento e a utilização de um banco de dados maior, visto que as ferramentas e tecnologias permanecem se desenvolvendo após esse estudo.

Além disso, é importante ressaltar que a validação do conceito permite a aplicação das ferramentas utilizadas para o desenvolvimento de um novo modelo para identificação e cálculo de peso de animais, percentual de gordura, posições de corte de peças, entre outras operações fabris que possuem potencial para serem automatizadas.

Apesar das limitações, o trabalho entregou um valor considerável de precisão e assertividade, considerando as 100 iterações realizadas de maneira otimizada com um conjunto de 189 imagens de treino, 55 imagens de validação e 27 imagens para testes. Os resultados obtidos indicam a viabilidade da aplicação da visão computacional no controle e monitoramento da produção em frigoríficos, abrindo precedentes para futuras implementações e otimizações em diferentes pontos da cadeia produtiva e em outros segmentos da indústria.

Uma das possíveis aplicações da abordagem estabelecida é a identificação de diferentes cortes ou partes de animais de forma a alimentar máquinas que permitam a automação de processos, tais como máquinas automáticas de desnuca, serras, extratores e outros equipamentos.

#### **4. REFERÊNCIAS**

ALIF, Mujadded Al Rabbani; HUSSAIN, Muhammad. YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain. [s. l.], p. 1–31, 2024. Disponível em: <http://arxiv.org/abs/2406.10139>.

ARAÚJO, Aline Moura. DETECÇÃO E DESTAQUE EM VÍDEO DE OBJETOS UTILIZANDO YOLO. **UNIVERSIDADE FEDERAL DA PARAÍBA**, [s. l.], p. 1–23, 2016.

BRAGA, Bernardo F. B. **Precisão e revocação**. [S. l.], 2017. Disponível em: [https://pt.wikipedia.org/wiki/Precisão\\_e\\_revocação#/media/Ficheiro:Precisão\\_e\\_revocação.png](https://pt.wikipedia.org/wiki/Precis%C3%A3o_e_revoca%C3%A7%C3%A3o#/media/Ficheiro:Precis%C3%A3o_e_revoca%C3%A7%C3%A3o.png) . .

CARRIJO GUIMARÃES, Alex *et al.* Estudo Referencial Bibliográfico Sobre Visão Computacional Na Indústria. [s. l.], n. Junho 2023, p. 1–8, 2023.

KIM, Haesik. Deep Learning. **Artificial Intelligence for 6G**, [s. l.], v. 22, n. 4, p. 247–303, 2022.

MARIANO, Diego. Métricas de avaliação em machine learning: acurácia, sensibilidade, precisão, especificidade e F-score. **BIOINFO - Revista Brasileira de Bioinformática e Biologia Computacional**, [s. l.], n. 2019, 2021.

MONTEIRO, Carlos *et al.* **Um sistema de baixo custo para reconhecimento de gestos em LIBRAS utilizando visão computacional**. [S. l.: s. n.], 2016.

PANDORFI, Héilton *et al.* Suinocultura de precisão: Visão computacional e tecnologias digitais. **Revista Científica de Produção Animal**, [s. l.], v. 22, n. 2, p. 73–79, 2020. Disponível em:  
<https://periodicos.ufpb.br/index.php/rcpa/article/view/57606>.

SANGOI, Luiz Fernando. ANÁLISES DO SISTEMA PRODUTIVO SUINÍCOLA DA REGIÃO CENTRAL DO RIO GRANDE DO SUL. **Procedia Manufacturing**, [s. l.], v. 1, p. 1–17, 2014.

ULTRALYTICS, © 2025. **Introducing Ultralytics YOLO**. [S. l.], 2025. Disponível em:  
<https://docs.ultralytics.com/> .

ULTRALYTICS. **Mergulho profundo na métrica de desempenho**. [S. l.], 2025. Disponível em: <https://docs.ultralytics.com/pt/guides/yolo-performance-metrics/#class-wise-metrics>. .