



**GERALDO LUCIANO DE CARVALHO NETO**

**SISTEMA INTELIGENTE DE VISÃO COMPUTACIONAL  
PARA ANÁLISE DE COMPORTAMENTO ALIMENTAR  
ANIMAL:  
APLICAÇÃO NA SUINOCULTURA**

**LAVRAS – MG**

**2021**

**GERALDO LUCIANO DE CARVALHO NETO**

**SISTEMA INTELIGENTE DE VISÃO COMPUTACIONAL PARA ANÁLISE DE  
COMPORTAMENTO ALIMENTAR ANIMAL:  
APLICAÇÃO NA SUINOCULTURA**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do curso de Engenharia de Controle e Automação, para obtenção do título de Bacharel.

Prof. DSc. Danilo Alves de Lima  
Orientador

**LAVRAS – MG  
2021**

**GERALDO LUCIANO DE CARVALHO NETO**

**SISTEMA INTELIGENTE DE VISÃO COMPUTACIONAL PARA ANÁLISE DE  
COMPORTAMENTO ALIMENTAR ANIMAL: APLICAÇÃO NA SUINOCULTURA  
COMPUTER VISION SMART SYSTEM FOR ANIMAL FEEDING BEHAVIOUR  
ANALYSIS: PIG FARMING CASE**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do curso de Engenharia de Controle e Automação, para obtenção do título de Bacharel.

APROVADA em 14 de Maio de 2021.

Prof. DSc. Wilian Soares Lacerda UFLA  
Prof. DSc. Arthur de Miranda Neto UFLA

Prof. DSc. Danilo Alves de Lima  
Orientador

**LAVRAS – MG  
2021**

## **AGRADECIMENTOS**

Primeiramente, agradeço aos meus pais por toda confiança em mim depositada ao me guiarem amorosamente por este caminho, fornecendo sempre apoio incondicional.

Agradeço a Universidade Federal de Lavras, universidade pública e de qualidade, que me permitiu trilhar esta realização profissional. A todos os servidores e colaboradores que fazem desta universidade uma das melhores do país.

Agradeço ao amigo Rodrigo de Lima, juntamente ao Núcleo de Estudos em Suinocultura (NESUI), do Departamento de Zootecnia da Universidade Federal de Lavras, pelo apoio fornecido na elaboração do problema, testes e validação do trabalho.

Aos meus amigos Leonardo e Matheus, por suas sinceras e leais amizades, que na universidade se iniciaram e que seguem para a vida.

A todos os respeitáveis professores e professoras que tive a oportunidade de receber valiosos ensinamentos.

Especialmente aos queridos professores Arthur de Miranda Neto e Danilo Alves de Lima, professores que muito admiro e me espelho, não somente profissionalmente, mas também pessoalmente. Meus sinceros agradecimentos por me proporcionarem o privilégio e honra de receber de vocês, mentorias que vão para além da academia.

Ao orientador deste trabalho, professor Danilo Alves de Lima, por sua paciência, dedicação e atenção.

Ao Núcleo do Laboratório de Mobilidade Terrestre (NLMT) onde tive ótimas convivências, amizades e importantes experiências práticas.

A Bruna, minha grande amiga e parceira de vida, por seu constante apoio e encorajamento.

Sobretudo, agradeço a Deus, por me conceder todas as bênçãos diárias no caminho desta conquista.

## RESUMO

Na produção de suínos, os custos com alimentação dos animais podem representar mais que 60% dos custos totais de produção. Análises de comportamento alimentar dos animais nestes ambientes de produção podem fornecer importantes informações para a produção eficiente e também para o bem-estar animal. Atualmente, tais análises são feitas por meio de observações humanas, demandando muita mão de obra e longo tempo, além de, geralmente, não possuírem resultados de grande confiabilidade. Com os recentes avanços das Redes Neurais Artificiais e das técnicas de Visão Computacional, o número de soluções tecnológicas inovadoras desenvolvidas com base nesses conhecimentos tem crescido exponencialmente. Suas aplicações são inúmeras e presentes nas mais diversas áreas. Este trabalho tem como objetivo propor um sistema que utiliza técnicas de Visão Computacional e Redes Neurais Convolucionais, uma classe de Rede Neural, para analisar, de forma automática, o comportamento alimentar de suínos em granjas, por meio de detecção e rastreamento visual dos animais. Para isso, uma rede neural Faster-R-CNN foi treinada para detecção de suínos, com uma precisão de 95,8% e revocação de 88,5%. A rede treinada se mostrou robusta, mesmo sob intensas variações de iluminação e sob oclusões parciais. O método proposto para análise de comportamento alimentar é capaz de registrar o tempo em que os animais se alimentam com uma precisão acima de 90%.

**Palavras-chave:** Aprendizado de Máquina. Visão Computacional. Suinocultura. Comportamento Animal. Redes Neurais Convolucionais.

## ABSTRACT

In pig production, the costs of feeding the animals can represent more than 60% of the total production costs. Feeding behavior analysis of animals in these production environments can provide important information for efficient production and also for animal welfare. Currently, such analyzes are made through human observations, demanding a lot of labor and a long time, besides, generally, they do not have accurate results. With the recent advances in Artificial Neural Networks and Computer Vision techniques, the number of innovative technological solutions developed based on this knowledge has grown exponentially. Its applications are numerous and present in the most diverse areas. This work aims to propose a system that uses Computer Vision techniques and Convolutional Neural Networks, a class of Neural Network, to automatically analyze the feeding behavior of pigs on farms, through the detection and visual tracking of animals. Thus, a Faster-R-CNN neural network was trained to detect pigs, with an accuracy of 95.8% and recall of 88.5%. The trained net proved to be robust, even under intense lighting variations and partial occlusions. The proposed method for the analysis of feeding behavior is capable of counting the time during which the animals feed with an accuracy above 90%.

**Keywords:** Machine Learning. Computer Vision. Pig Farming. Animal Behaviour. Convolutional Neural Networks.

## LISTA DE FIGURAS

Figura 3.1 – Ilustração em alto nível de uma arquitetura básica de Rede Neural Convolutiva, contendo Camadas de Convolução, Pooling e Camadas Totalmente Conectadas . . . . .	22
Figura 3.2 – Ilustração de operações de camadas de convolução em Redes Neurais Convolucionais: a partir da convolução entre um mapa de entrada, $X$ e bancos de filtros, $W$ , são obtidas as saídas das camadas de convolução, $O$ . São considerados mapas de entrada com uma profundidade 3, representando os componentes R, G e B de uma imagem. A janela de convolução representada "desliza" pelo mapa de entrada, computando o mapa de saída. . . . .	23
Figura 3.3 – Funcionamento de operações de Pooling em Redes Neurais Convolucionais.	24
Figura 3.4 – Ilustração do funcionamento de uma Rede Neural Convolutiva básica: O operador de convolução $\otimes$ indica a convolução 2D de uma imagem ou mapa de características com um Banco de Filtros. O passo denotado por (C) diz respeito à convolução. O passo denotado por (MP) indica uma operação de Pooling de Máximo. O Classificador está sendo omitido da arquitetura, sendo representadas somente as operações que realizam a extração de características. . . . .	25
Figura 3.5 – Funcionamento do Detector de Objetos R-CNN: A partir de uma imagem de entrada (1) são extraídas 2000 propostas de região (2). Cada uma das regiões são alimentadas em uma ConvNet para extração de características (3) que são utilizadas para a classificação da região (4). . . . .	27
Figura 3.6 – Funcionamento do Detector de Objetos Faster-R-CNN: A imagem de entrada é alimentada à Rede Neural Convolutiva de extração de características. O Mapa de Características é então utilizado como entrada para a Rede RPN que identifica Propostas de Regiões de objetos. Essas regiões são então redimensionadas por meio de uma camada de pooling de região de interesse (RoI). Por fim, essas regiões são classificadas em classes de objetos. . . . .	28

Figura 3.7 – Ilustração do funcionamento da Region Proposal Network: a partir de cada posicionamento de uma janela que percorre o mapa de características, são geradas $k$ caixas âncora, definidas por suas coordenadas e pontuações, que podem então ser selecionadas como propostas de região. . . . .	29
Figura 3.8 – a) Módulo <i>Inception</i> sem redução de filtros e b) Módulo <i>Inception</i> com redução de filtros . . . . .	30
Figura 3.9 – Exemplos de estratégias de Módulos <i>Inception</i> Versão 2. a) Fatoração $5 \times 5$ em duas $3 \times 3$ e b) Fatoração $1 \times n$ e $n \times 1$ . . . . .	31
Figura 4.1 – Exemplo de imagens rotuladas presentes no banco de dados COCO. . . . .	33
Figura 4.2 – Exemplo de imagem utilizada para o treinamento e teste da rede neural convolucional. . . . .	34
Figura 4.3 – Diagrama da solução de detecção de rastreamento de suínos, incluindo registro de comportamento alimentar. (1) Aquisição da imagem de entrada, (2) Detecção dos animais via Rede Convolucional Faster-R-CNN, (3) Rotulação dos indivíduos e (4) Registro de tempo e frequência alimentar. . . . .	36
Figura 4.4 – Ilustração do posicionamento ideal de câmera para o rastreamento dos animais. . . . .	37
Figura 4.5 – Área de intersecção entre detecções de quadros consecutivos. . . . .	38
Figura 4.6 – Diagrama da solução de detecção de rastreamento de suínos com a exemplificação de um caso real com dois quadros consecutivos. (a) Amostragem de uma imagem de entrada, (b) Resultado da detecção via Rede Convolucional, (c) Resultado da identificação de cada indivíduo e (d) Registro de tempo e frequência alimentar. . . . .	40
Figura 5.1 – Cenários considerados para os testes. . . . .	42
Figura 5.2 – Interface de Usuário desenvolvida para o sistema. . . . .	43
Figura 5.3 – Resultados do detector em diferentes cenários. . . . .	45

## LISTA DE TABELAS

Tabela 5.1 – Tabela verdade de detecção de animais para o conjunto de teste. . . . .	42
Tabela 5.2 – Precisão de Revocação do detector de objetos treinado. . . . .	43
Tabela 5.3 – Tabela de tempo de alimentação (em segundos) para o Cenário 1. . . . .	46
Tabela 5.4 – Tabela de tempo de alimentação (em segundos) para o Cenário 2. . . . .	46

## SUMÁRIO

<b>1</b>	<b>Introdução . . . . .</b>	<b>10</b>
<b>1.1</b>	<b>Objetivos . . . . .</b>	<b>11</b>
<b>1.2</b>	<b>Justificativa . . . . .</b>	<b>11</b>
<b>1.3</b>	<b>Estrutura do Trabalho . . . . .</b>	<b>12</b>
<b>2</b>	<b>Revisão da Literatura . . . . .</b>	<b>13</b>
<b>2.1</b>	<b>Automação na criação animal . . . . .</b>	<b>13</b>
<b>2.2</b>	<b>Automação aplicada à suinocultura . . . . .</b>	<b>14</b>
<b>2.3</b>	<b>Identificação individual e a alimentação de precisão na suinocultura . . . . .</b>	<b>15</b>
<b>2.4</b>	<b>Deteção e rastreamento visual de suínos . . . . .</b>	<b>16</b>
<b>3</b>	<b>Conceitos Gerais . . . . .</b>	<b>18</b>
<b>3.1</b>	<b>Deteção e rastreamento de Objetos . . . . .</b>	<b>18</b>
<b>3.1.1</b>	<b>Deteção de objetos . . . . .</b>	<b>18</b>
<b>3.1.2</b>	<b>Rastreamento de Objetos . . . . .</b>	<b>19</b>
<b>3.2</b>	<b>Redes Neurais Convolucionais . . . . .</b>	<b>21</b>
<b>3.3</b>	<b>Resultados Qualitativos . . . . .</b>	<b>22</b>
<b>3.3.1</b>	<b>Treinamento de Redes Neurais Convolucionais . . . . .</b>	<b>24</b>
<b>3.3.2</b>	<b>Redes Neurais Convolucionais para Deteção de Objetos . . . . .</b>	<b>26</b>
<b>3.3.3</b>	<b>Arquitetura Inception de Rede Neurais Convolucionais . . . . .</b>	<b>28</b>
<b>3.3.3.1</b>	<b>Inception versão 1 . . . . .</b>	<b>29</b>
<b>3.3.3.2</b>	<b>Inception Versão 2 . . . . .</b>	<b>30</b>
<b>3.4</b>	<b>Desempenho de detectores de objetos . . . . .</b>	<b>31</b>
<b>4</b>	<b>Metodologia . . . . .</b>	<b>33</b>
<b>4.1</b>	<b>Treinamento da Rede Neural Convolucional para Deteção de Suínos . . . . .</b>	<b>33</b>
<b>4.2</b>	<b>Diagrama geral da solução de deteção e rastreamento . . . . .</b>	<b>35</b>
<b>4.3</b>	<b>Aquisição da Imagem . . . . .</b>	<b>36</b>
<b>4.4</b>	<b>Deteção dos animais via Rede Neural Convolucional . . . . .</b>	<b>37</b>
<b>4.5</b>	<b>Identificação dos Indivíduos . . . . .</b>	<b>38</b>
<b>4.6</b>	<b>Registro de Comportamento Alimentar . . . . .</b>	<b>39</b>
<b>5</b>	<b>Resultados . . . . .</b>	<b>41</b>
<b>5.1</b>	<b>Treinamento da Rede Neural Convolucional para Deteção de Suínos . . . . .</b>	<b>41</b>
<b>5.2</b>	<b>Interface de Usuário . . . . .</b>	<b>42</b>

<b>5.3</b>	<b>Resultados Qualitativos . . . . .</b>	<b>44</b>
<b>5.4</b>	<b>Resultados Quantitativos . . . . .</b>	<b>44</b>
<b>6</b>	<b>Considerações Finais . . . . .</b>	<b>47</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>48</b>

## 1 INTRODUÇÃO

A carne suína é mais consumida mundialmente, com aproximadamente 770 milhões de porcos consumidos todo ano, resultando na produção de mais de 110 milhões de toneladas de carne (USDA, 2017). Ao contrário da criação em confinamento individual, a criação de suínos em baias coletivas proporciona maior bem estar ao animal, além de melhores resultados de produtividade, uma vez que reduz o estresse do animal e pode elevar a qualidade da carne (NETO; NOBRE, 2015). Esse sistema de criação, no entanto, requer maior volume de trabalho por parte dos produtores, uma vez que as tarefas de localização e identificação dos animais passam a ser um desafio.

Localizar e identificar os animais são atividades desafiadoras em ambientes grandes de produção, mas são essenciais pra monitorar a saúde do animal e mensurar a produtividade da granja, por meio de cálculos de conversão de peso, por exemplo. Elas são, também, fundamentais em situações experimentais, onde são testadas diferentes rações e métodos de manejo animal. Tais tarefas, atualmente, são executadas por meio de observação humana, que não só consomem tempo, mas também podem levar a observações imprecisas, uma vez que os animais são conhecidos por mudar ou cessar comportamentos de interesse na presença de um observador humano (LERUSTE et al., 2013).

Para tentar reduzir essa carga de trabalho e aumentar a produtividade e precisão, alguns métodos de identificação e rastreamento foram propostos no passado, como: a identificação por meio de marcas de tinta ou via utilização de tecnologia RFID (MASELYNE et al., 2014b). Seja por meio de aplicação de tinta, ou pela inserção de etiquetas eletrônicas nos animais, os métodos citados exigem trabalho manual periódico para inserção e manutenção de identificadores nos animais. Tal manutenção se torna bastante laboriosa e recorrente quando se trata de suínos, uma vez que esses tendem a interagir entre si com grande frequência, removendo as etiquetas um do outro, ou apagando os marcos de tinta pelo atrito. Além disto, apesar de alguns sistemas de identificação e rastreamento por radiofrequência terem mostrado bons resultados (VOULODIMOS et al., 2010), a solução costuma ter alto custo de implementação, pois também envolve a aquisição e instalação de antenas de longo alcance e de alimentadores eletrônicos com identificadores.

Com a utilização de técnicas de visão computacional, foram propostos diferentes métodos que permitem rastrear animais sem a necessidade de utilização de etiquetas ou marcas, reduzindo trabalhos manuais e manutenções. Tais métodos, no entanto, tendem a ter sua acurácia fortemente dependente do ambiente ou *background* onde são aplicados (AHRENDT; GRE-

GERSEN; KARSTOFT, 2011). Alguns métodos mais resilientes foram desenvolvidos, porém exigem o uso de câmeras de calor ou câmeras estéreo (KIM et al., 2017), que possuem custo elevado. Câmeras monoculares, que são câmeras comuns, oferecem uma alternativa de baixo custo, mas exigem técnicas de visão computacional que sejam robustas a mudanças de iluminação e ambiente.

A proposta deste trabalho é desenvolver um sistema de detecção e rastreamento de suínos por meio de visão computacional, de forma resiliente ao ambiente, e que utilize somente câmeras comuns (monoculares). Para isto, foi desenvolvido uma rotina que utiliza rede convolucional para a detecção dos animais e métodos de rastreamento visual de objetos para acompanhar seus movimentos, fornecendo informações precisas à granja, com um baixo custo e pouca mão de obra.

## **1.1 Objetivos**

O objetivo deste trabalho é propor um sistema de detecção e rastreamento de suínos por meio de visão computacional, de forma resiliente ao ambiente, e que utilize somente câmeras comuns (monoculares). Este sistema deve ser capaz de fornecer informações precisas à granja, com um baixo custo e pouca mão de obra. Para isto, os seguintes itens serão desenvolvidos e apresentados:

1. Definição e treinamento de uma rede convolucional (CONV-NET) para detecção em tempo real de suínos;
2. Concepção de um método de rastreamento espaço-temporal dos animais para extração de comportamento;
3. Desenvolvimento de uma interface homem-máquina onde é possível realizar o processo de seleção de vídeo e gerar relatórios.

## **1.2 Justificativa**

O monitoramento constante do comportamento dos animais permite ao produtor de carne suína identificar, de forma precoce, animais doentes, realizar o balanço de conversão de peso, além de gerar informações comportamentais que dizem respeito ao bem-estar animal. Esses são fatores importantes para promover maior eficiência, competitividade e bem-estar animal no ambiente de criação. O método proposto neste trabalho tem o potencial de automatizar

esse processo de monitoramento e, se comparado a métodos semelhantes presentes na literatura, entregar uma operação mais robusta e independente do ambiente de operação ou de equipamentos especializados, como: leitores e etiquetas de radiofrequência ou câmeras térmicas. Essa é uma grande vantagem, pois incorre em menores custos e maior precisão de operação. Este trabalho também poderá ser utilizado como ponto de partida para outras pesquisas que pretendam otimizar e melhorar os sistema de identificação e *tracking* aqui expostos.

### 1.3 Estrutura do Trabalho

Este documento está dividido em seis capítulos. O primeiro capítulo fez uma breve contextualização e a problemática foi apresentada ao leitor. Nesse mesmo capítulo também são apresentados argumentos iniciais que justificam o trabalho.

No Capítulo 2 é apresentado um histórico da utilização de aprendizado de máquina e da visão computacional para identificação e rastreamento de objetos por meio de imagem. Nesse sentido, são expostos trabalhos semelhantes ao aqui apresentado e/ou que utilizam tais ferramentas de formas semelhantes.

De forma mais detalhada, as ferramentas e métodos básicos utilizados para elaborar e executar o método aqui proposto são descritos no Capítulo 3. Essas ferramentas são: Elaboração e treinamento de Redes Convolucionais e rastreamento visual por comparação espaço-temporal.

O Capítulo 4 é uma exposição específica da metodologia desenvolvida neste trabalho, bem como o diagrama da solução e detalhes de implementação. Os resultados quantitativos e qualitativos dos testes executados pela metodologia proposta são descritos no Capítulo 5, fazendo uma análise com demais métodos.

Por fim, o Capítulo 6 apresenta as considerações finais e propostas de melhoria são feitas, para nortear possíveis trabalhos futuros.

## 2 REVISÃO DA LITERATURA

A automação é o uso de equipamento mecânico e/ou eletrônico para reduzir a necessidade de trabalho humano (EVODAREK., 2012). Devido ao grande potencial de gerar notáveis benefícios ao produtor, os efeitos econômicos do emprego de sistemas de automação na indústria de produção animal já vêm sendo discutidos há décadas (NOTT, 1979). Atualmente, o emprego dessas tecnologias traz grandes vantagens, dentre elas, a redução de custos de produção e a maximização do lucro. Além disso, o fato de a cada ano o consumo de produtos de origem animal aumentar 7% (GUPTA et al., 2015), tornou a aplicação de tecnologia no campo uma necessidade crescente.

### 2.1 Automação na criação animal

Sistemas de automação para a produção animal vêm sendo empregados de diversas formas: desde sistemas de simples registro de dados da granja, que auxiliam na gestão e tomada de decisão, até o uso de equipamentos que controlam e monitoram o consumo de alimento e temperatura individual do animal (NOTT, 1979). Na criação de aves, por exemplo, incubatórios automatizados aumentam a produção e garantem a saúde dos filhotes, além de contar, embalar, pesar os ovos e filhotes, e remover os ovos de má qualidade.

Em culturas de criação extensiva, onde o rebanho é criado em ambiente aberto, a abordagem é feita de modo diferente daquela onde o animal é confinado, e existem, por exemplo, soluções aplicadas a criação de gado que utilizam colares com GPS (Global Positioning System) e sensores inerciais capazes de armazenar a localização do animal com alta frequência temporal. Os dados armazenados são coletados por meio de redes sem fio e combinadas com imagens de satélite. As informações geoespaciais combinadas àquelas de localização e movimentação, permitem inferir a preferência de vegetação do animal, prevenindo a degradação descontrolada de partes específicas do pasto.

A interação entre os animais pode notificar, por exemplo, eventos de acasalamento, permitindo analisar não só o comportamento do animal, mas também a sua interação com o ambiente e os demais animais (HANDCOCK et al., 2009). Esse tipo de informação é de grande importância, uma vez que permite ao produtor determinar as estratégias ótimas de intervenção no manejo (TURNER et al., 2009).

Tecnologias similares utilizam modelos para analisar os dados de sensores inerciais inseridos nos colares do gado e detectar comportamentos que indicam a fertilidade e possíveis doenças no animal (MOTTRAM, 2016).

Quando se trata da criação de gado em confinamento, é possível encontrar soluções que utilizam técnicas de visão computacional para analisar o comportamento do animal, como apresentado em (PORTO et al., 2013), onde imagens de câmeras de monitoramento são processadas por meio de um algoritmo que detecta e classifica automaticamente o comportamento do gado ao deitar-se. Essa classificação é relevante ao passo que mudanças nesse comportamento podem indicar problemas físicos e sociais que impactam negativamente na saúde e na eficiência reprodutiva do gado.

Na pecuária de corte, a mensuração do ganho de peso do animal é uma das informações mais relevantes, pois indica o desempenho do animal, fornecendo uma perspectiva de resultado ao pecuarista. Isto torna a aferição de peso dos animais uma tarefa recorrente. Esta tarefa, no entanto, incorre em alto nível de estresse ao animal, pois é geralmente feita de forma manual, utilizando balanças de pesagem convencionais. Com o intuito de automatizar este processo, reduzindo mão de obra e poupando o animal de impactos negativos decorrentes do estresse, a visão computacional pode se tornar o pilar de soluções que propõe inferir o peso do gado de corte à distância. Em (COMINOTTE et al., 2020), por exemplo, é analisada a aplicação de um sistema de visão computacional que permite prever o peso corporal e o ganho diário de peso de gados de corte na fase de ganho e terminação, sem nenhum contato, utilizando câmeras estéreo de baixo custo.

## **2.2 Automação aplicada à suinocultura**

Na suinocultura, experimentos comprovam que condições ambientais (temperatura, umidade, velocidade do ar, etc.) influenciam no conforto e comportamento do animal e, consequentemente, no crescimento, desenvolvimento, taxa de conversão alimentar, e saúde (MOUNT, 1968), (VESTEGEN BRASCAMP, 1978), (HESSING; TIELEN, 1994).

Partindo destas comprovações, existem no mercado vários sistemas de controle de ambiente que utilizam sensores de temperatura, ventilação, umidade e gases para medir a qualidade do ambiente e atuar na ventilação ou no aquecimento do ambiente. Isso permite obter, de forma automática, um ambiente ótimo para o bem-estar e criação, resultando no maior desempenho do animal.

A automação na suinocultura, no entanto, tem avançado, propondo uma automatização que vai além da climatização do ambiente, e que passa a fazer parte da operação das granjas, tornando-a cada vez mais precisa e eficiente. Isso acontece por meio da automatização de tarefas chave do processo como, por exemplo, a identificação individual dos animais, a pesagem automática e individual, e com a alimentação de precisão.

### **2.3 Identificação individual e a alimentação de precisão na suinocultura**

Em zonas de produção industrial e semi-industrial de suínos, os custos com alimentação representam mais que 60% dos custos totais da produção (POMAR et al., 2009), o que torna a aplicação de técnicas de alimentação de precisão um boa estratégia para maximizar a eficiência produtiva. A alimentação de precisão é aquela onde a quantidade e concentração do alimento de cada animal é controlado de forma individual, de acordo com suas necessidades, o que eminentemente requer a identificação individual de cada suíno.

Em granjas de confinamento, utilizadas geralmente na maternidade, essa identificação pode ser feita por meio do uso de códigos QR (Quick Response), como propõem (CHEN et al., 2016) e (PARK; JEONG; YOE, 2014). Nessa aplicação, a marcação de cada animal é um grande desafio, principalmente quando em baias coletivas de criação, pois, assim como em estratégias rudimentares, que utilizam de tinta para marcar os animais, o processo se torna bastante ineficiente e exige mão de obra recorrente, uma vez que os animais tendem a retirar com facilidade, por meio de atrito com o ambiente e os demais animais, qualquer marcação neles aplicada.

Para automatizar esse processo de identificação, é comumente adotado o uso de implantes subcutâneos ou brincos com etiquetas RFID nos animais. O uso da tecnologia permite a identificação do indivíduo por meio de leitores de proximidade. Esta informação de identificação, aliada à equipamentos apropriados, pode ser utilizada com a finalidade de implementar a alimentação de precisão (MASELYNE et al., 2014a), permitindo também medir o consumo de água de cada suíno (MASELYNE et al., 2016). No entanto, a tecnologia possui restrições de distância, exigindo proximidade para a leitura e, para que este tipo de automação possa ser utilizada, é necessário a instalação de uma infraestrutura apropriada com alimentadores individuais dotados de antenas leitoras RFID, além do implante e manutenção das etiquetas nos animais, o que geralmente incorre em altos custos, não se tornando acessível a qualquer ambiente de produção.

Com o avanço de técnicas de visão computacional, novas propostas da literatura têm usado da tecnologia no âmbito da suinocultura. A identificação individual pode ser feita por meio de reconhecimento facial do animal, como faz (HANSEN et al., 2018). O uso de técnicas de visão também abre novas possibilidades de extração de informações de grande relevância, como a de locomoção do grupo de animais dentro da granja, a distância entre os suínos, e suas interações físicas. Tais informações comportamentais são fortes indicadores do estado de bem-estar do animal (PERNER, 2001). De forma específica, a locomoção, ou a falta dela, pode evidenciar situações de desconforto térmico (SHAO; XIN, 2008), estresse, dor, e possíveis doenças ou lesões, como a claudicação<sup>1</sup> (ANIL; ANIL; DEEN, 2002). A literatura mostra que animais estressados ou com dor podem apresentar menores índices de crescimento, produção e reprodução (LAUBER; VICTORIA., 2007). À vista disso, entende-se que o estado de bem-estar dos animais é de grande valia para o produtor, não somente do ponto de vista ético, mas também econômico, uma vez que o bom manejo de bem-estar animal resulta em bons resultados em termos de eficiência produtiva, qualidade e comerciabilidade (LAUBER; VICTORIA., 2007).

#### **2.4 Detecção e rastreamento visual de suínos**

Métodos de visão computacional para detecção ou rastreamento de locomoção geralmente se baseiam em identificação e subtração de regiões de fundo (o ambiente), de regiões com possível presença do animal. Usualmente, esse processo é feito mediante o uso de algoritmos baseados em um limiar da escala de cinza (TORELLO et al., 1983), modelos estatísticos (TILLET; ONYANGO; MARCHANT, 1997), ou por detecção de cor (NOLDUS; SPINK; TEGELENBOSCH, 2001). Por serem métodos fortemente dependentes de cor, iluminação e ambiente de fundo, dependem de ajustes finos e tendem a ter resultados comprometidos quando alguma destas variáveis se diferem do ambiente inicialmente testado (LIND et al., 2005).

Para enfrentar essa deficiência intrínseca, novos métodos propõem o uso de câmeras térmicas (KIM; CHO; LEE, 2017) ou câmeras estéreo aliadas a algoritmos de percepção de profundidade (KIM et al., 2017), para mais fácil subtração dos animais da imagem. Enquanto os citados métodos obtêm bons resultados, o uso de câmeras especiais, diferentes das comuns monoculares, incidem em custos elevados e, conseqüentemente, inviáveis para a maioria dos ambientes de produção. Levando em conta as dificuldades supracitadas, o método proposto

---

<sup>1</sup> Claudicação se refere a uma marcha ou postura anormal de um animal que é o resultado de uma disfunção do sistema locomotor.

neste trabalho tem o objetivo de executar a tarefa de rastreamento de animais em granjas com o uso de câmeras monoculares, e utilizar como estratégia de detecção de porcos, o uso de um modelo de Rede Neural Convolucional (INDOLIA et al., 2018) mais robusto a mudanças de ambiente de fundo, condições de luminosidade, ou cor.

### 3 CONCEITOS GERAIS

Neste capítulo serão contextualizados os conceitos fundamentais acerca do tema do trabalho. Por este se tratar de uma abordagem que utiliza como alicerce métodos de visão computacional para a detecção e rastreamento de objetos, esses dois tópicos serão os primeiros a serem conceituados. Posteriormente, as técnicas de detecção e rastreamento de objetos utilizadas neste trabalho serão explicadas em detalhes.

Uma vez que a proposta deste trabalho permite o rastreamento de múltiplos objetos de forma simultânea, torna-se importante descrever o conceito de MOT (Multiple Object Tracking). Por fim, para a detecção de cada objeto, é utilizada uma técnica que se baseia em aprendizado de máquina, um conceito de grande relevância que será elucidado em seguida.

#### 3.1 Detecção e rastreamento de Objetos

Devido ao grande número de aplicações acadêmicas e comerciais, a detecção e rastreamento visual de objetos em movimento se tornou um importante tema. Sua aplicação pode ser encontrada em diferentes domínios, como na robótica, sendo parte fundamental de sistemas navegação autônoma de robôs e veículos (ESS et al., 2010), em softwares de segurança, sistemas de monitoramento por vídeo (JOSHI; THAKORE, 2012) e até mesmo aplicações para fins militares (KAMATE; YILMAZER, 2015). Nesta seção, alguns conceitos básicos de detecção e rastreamento visual de objetos serão apresentados.

##### 3.1.1 Detecção de objetos

A detecção de um objeto em uma imagem consiste em verificar a presença de um ou mais objetos semânticos pertencentes à uma determinada classe (como pessoas, animais, veículos ou prédios) (DASIOPOULOU et al., 2005). Desse modo, a detecção de objetos permite estimar em uma imagem regiões de interesse que contêm os objetos semânticos buscados. Essa detecção pode ser executada de diversas maneiras. Soluções mais simples inferem a localização desses objetos por meio da identificação de características definidas pelo desenvolvedor. São características básicas, inerentes a tais objetos, bordas e cantos (HARRIS; STEPHENS, 1988), formas (DASIOPOULOU et al., 2005) e cores (NOLDUS; SPINK; TEGELENBOSCH, 2001).

Abordagens de detecção baseadas em características manualmente definidas, por dependerem de ajustes finos, são bastante suscetíveis a erros, uma vez que alterações de iluminação

podem ofuscar as características do objeto ou até mesmo as características do próprio ambiente de fundo podem se tornar falsos positivos, frente à essas alterações (PAPAGEORGIOU; POGGIO, 2000). Devido a essas dificuldades, novos métodos passaram a utilizar técnicas de aprendizado de máquina para que a extração de características fosse feita de forma mais automática e completa. Nesse sentido, métodos como o "Viola Jones" (VIOLA; JONES, 2001) e o detector "HOG" (DALAL; TRIGGS, 2005) foram apresentados, obtendo resultados de notável impacto por sua velocidade de processamento, resiliência à mudanças de cenário e baixa dependência de ajustes finos.

Resultados disruptivos na área vieram com o renascimento da rede neural convolucional (ZOU et al., 2019), por sua capacidade de aprender a representar características de alto nível de complexidade presentes em uma imagem. Uma análise mais aprofundada acerca de redes neurais convolucionais aplicadas à detecção de objetos será apresentada na Subseção 3.3.2.

### **3.1.2 Rastreamento de Objetos**

O rastreamento de objetos, ou rastreamento de vídeo, consiste em localizar um (ou múltiplos) objeto ao longo do tempo, em uma sequência de imagens, construindo sua trajetória no plano da imagem. Tradicionalmente, o problema do rastreamento é formulado como uma estimativa recursiva sequencial, tendo uma estimativa da distribuição de probabilidade da localização do objeto na imagem anterior. O problema é estimar a distribuição de probabilidade na nova imagem, usando todo o conhecimento previamente disponível e as novas informações advindas da nova imagem (ABDELALI; ESSANNOUNI; ABOUTAJDINE, 2016).

As tarefas de detectar o objeto e de estabelecer uma correspondência entre as detecções na sequência de imagens podem ser executadas tanto separadamente quanto de modo conjunto. No primeiro caso, possíveis regiões de objetos são detectadas a cada imagem, individualmente, por meio do algoritmo de detecção. No segundo caso, a região onde se encontra o objeto e a correspondência na imagem seguinte são estimadas de forma concomitante, atualizando iterativamente as informações de localização e região do objeto obtidas da imagem anterior (YILMAZ; JAVED; SHAH, 2006).

Em (YILMAZ; JAVED; SHAH, 2006), é ressaltado que a grande complexidade em se rastrear um objeto em uma sequência de imagens pode advir de, principalmente:

- Perda de informação causada pela projeção do mundo 3D no plano 2D da imagem,
- Ruído nas imagens,

- Movimento complexo dos objetos,
- Natureza não-rígida ou articulada de objetos,
- Oclusão parcial ou total do objeto,
- Formato complexo de objetos,
- Mudanças de iluminação da cena,
- Necessidade de processamento em tempo real.

Sendo assim, o nível de complexidade do algoritmo é definido por características relacionadas à aplicação para a qual o método é proposto. Com o intuito de reduzir essa complexidade pode ser suposto, por exemplo, que o objeto não muda de forma, ou que o objeto se movimenta linearmente e em baixas velocidades. Isso pode simplificar o esforço e nível de complexidade do algoritmo de rastreamento.

A maior parte das técnicas de rastreamento propostas na literatura podem ser categorizadas como Rastreamento de Kernel<sup>1</sup>, de Silhueta ou de Ponto. O primeiro se refere ao rastreamento que utiliza como parâmetros mudanças nos pixels contidos em uma região primária pré-selecionada que contenha o objeto. Essa região pode ser, por exemplo, um retângulo ou uma elipse. Dentre os métodos propostos nesse sentido, se destacam os métodos que utilizam a correspondência de modelo (Template Matching), que procura na imagem uma região similar a região selecionada na imagem anterior (COMANICIU et al., 2003), e técnicas que utilizam fluxo óptico da região para estimar o movimento do objeto (SHI; TOMASI, 1994).

O Rastreamento de Silhueta parte da necessidade de descrever o contorno do objeto de forma mais detalhada, com formas mais complexas. Esses algoritmos geralmente executam tanto a detecção quanto o rastreamento do objeto, uma vez que, em sua maioria, são pré-treinados com um modelo que representa a silhueta do objeto a ser rastreado. Yilmaz, Javed e Shah (2006) definem que Rastreadores de Silhueta podem ser divididos em duas categorias, rastreadores por correspondência de silhueta e de contorno. Aqueles por correspondência de silhueta, buscam a silhueta do objeto na imagem atual, enquanto os de contorno evoluem um contorno inicial do objeto para sua nova posição na imagem atual.

Já o rastreamento de pontos consiste em encontrar a correspondência entre pontos de detecção de objetos em imagens consecutivas. Esses algoritmos partem do pressuposto de que

---

<sup>1</sup> O termo Kernel se refere ao núcleo ou parte central do objeto de interesse.

há uma técnica à parte, utilizada para realizar a detecção dos objetos a cada imagem e utilizam de métodos determinísticos ou estatísticos para inferir a correspondência entre detecções, estimando o movimento do objeto. Métodos determinísticos assumem algumas restrições acerca do movimento do objeto, como velocidade máxima, proximidade de localização em relação a imagem anterior, e pequena mudança de movimento para minimizar o custo de associação de um objeto em uma imagem em relação a algum objeto presente na imagem seguinte (VEENMAN; REINDERS; BACKER, 2001). No entanto, essas restrições também podem ser assumidas para métodos estatísticos.

Métodos estatísticos de rastreamento de ponto levam em conta as incertezas da estimativa do estado do objeto, como ruído e perturbação na movimentação do mesmo. Nesse sentido, o filtro de Kalman (WELCH; BISHOP et al., 1995) é bastante utilizado.

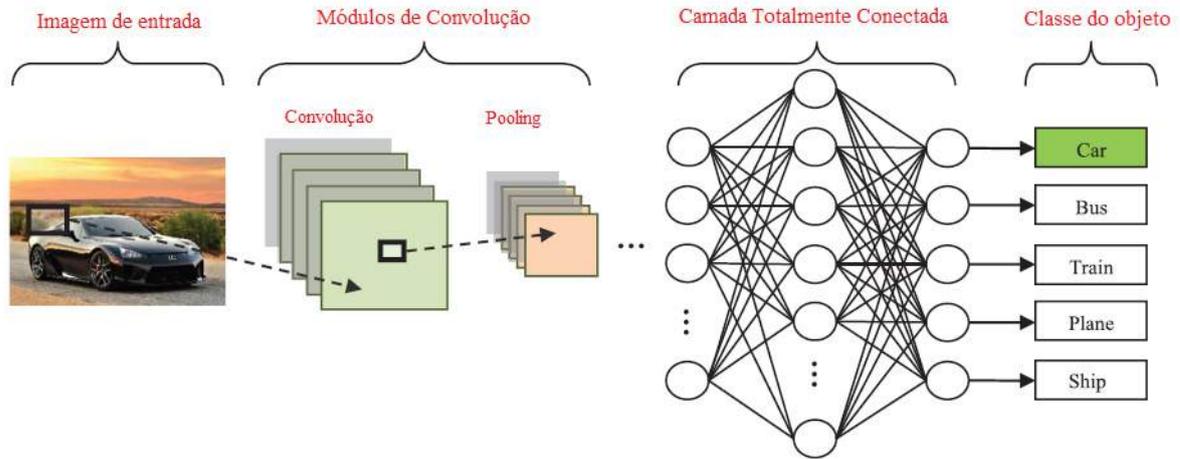
O método implementado nesse trabalho para o rastreamento dos animais utiliza conceitos de rastreamento de Kernel e também de rastreamento de Ponto para realizar a associação dos animais detectados ao longo da sequência de imagens, compondo as trajetórias dos mesmos. O algoritmo desenvolvido para esse fim será descrito em detalhes no Capítulo 4.

### 3.2 Redes Neurais Convolucionais

Redes Convolucionais são um tipo especializado de redes neurais artificiais que usam a operação de convolução no lugar da multiplicação de matrizes em pelo menos uma de suas camadas. Existem inúmeras variações de arquiteturas de Redes Neurais Convolucionais, mas, em geral, elas consistem em um sequência de Camadas de Convolução e Camadas de Pooling (ou subamostragem), que são agrupadas em módulos. Em seguida, após várias camadas dessas, há pelo menos uma camada totalmente conectada que realiza a classificação final (RAWAT; WANG, 2017) e é, portanto, chamado de Classificador. A Figura 3.1 é uma ilustração em alto nível de uma arquitetura básica de Rede Neural Convolucional.

As camadas de convolução são utilizadas para extrair características e, para tanto, aprendem, por meio de um algoritmo de aprendizado de máquina, diferentes representações de características dos objetos das imagens apresentadas (RAWAT; WANG, 2017). Nessas camadas ocorre nada mais do que a convolução, simbolizada pelo operador  $\otimes$ , do mapa de entrada (ou imagem de entrada),  $X$  com um banco de filtros  $W$ , que consiste em um conjunto de pesos. O resultado dessa convolução é utilizado como entrada para uma função de ativação não-linear  $f$  que tem como saída um mapa de características  $Y$ . Sendo assim, o  $k$ -ésimo mapa de caracterís-

Figura 3.1 – Ilustração em alto nível de uma arquitetura básica de Rede Neural Convolutiva, contendo Camadas de Convolução, Pooling e Camadas Totalmente Conectadas



Fonte: (RAWAT; WANG, 2017)

tics de saída  $Y_k$  pode ser definido como

$$Y_k = f(W_k \otimes X), \quad (3.1)$$

onde a função de ativação não-linear utilizada é tradicionalmente uma função Sigmoid ou, mais recentemente convencionado, a função ReLU<sup>2</sup>. A Figura 3.2 ilustra as operações que ocorrem nas camadas de Convolução.

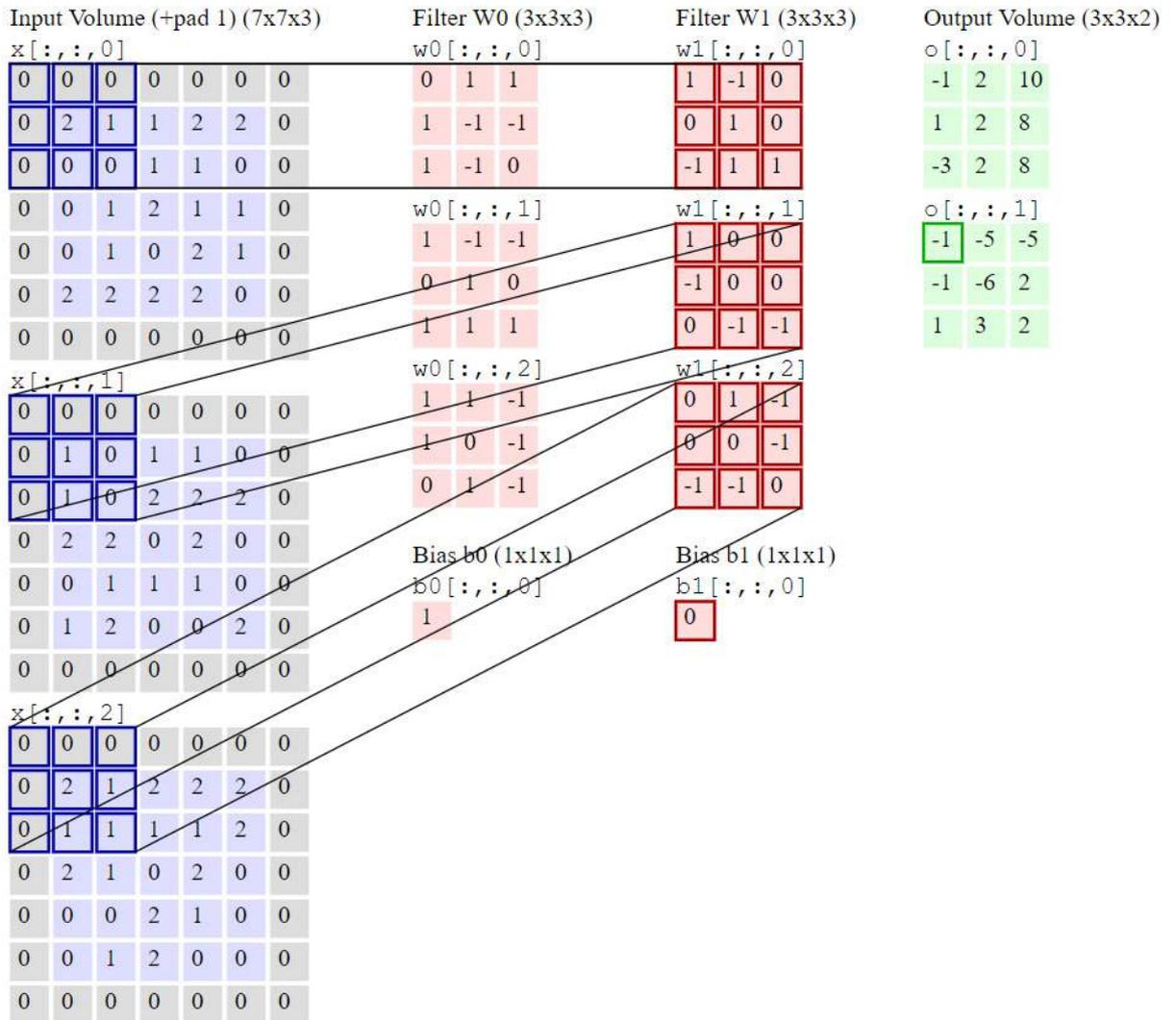
As camadas de Pooling, ou Camadas de Sub-amostragem, reduzem a resolução espacial de cada mapa de características. Isso é feito por meio da separação do mapa em sub-regiões não sobrepostas de determinado tamanho e a extração de um único valor que melhor representa cada uma dessas sub-regiões. Esse valor é utilizado como saída, tornando-se parte do mapa de saída da operação de Pooling. Sua função é reduzir o número de parâmetros da rede a fim de reduzir o esforço computacional exigido pela rede e também evitar situações de sobreajuste. A Figura 3.3 exemplifica o funcionamento de uma operação de Pooling.

### 3.3 Resultados Qualitativos

A determinação desse valor de saída pode ser feita de diferentes técnicas. As mais comuns são calculando a média dos valores da sub-região, chamada de Pooling de Média, ou

<sup>2</sup> O termo ReLU se refere a função "Rectified Linear Units" (NAIR; HINTON, 2010).

Figura 3.2 – Ilustração de operações de camadas de convolução em Redes Neurais Convolucionais: a partir da convolução entre um mapa de entrada,  $X$  e bancos de filtros,  $W$ , são obtidas as saídas das camadas de convolução,  $O$ . São considerados mapas de entrada com uma profundidade 3, representando os componentes R, G e B de uma imagem. A janela de convolução representada "desliza" pelo mapa de entrada, computando o mapa de saída.

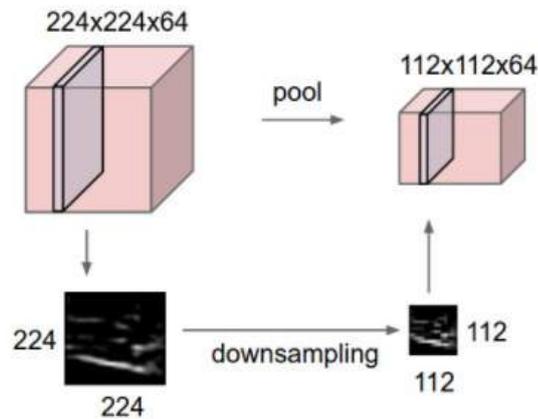


Fonte: (KARPATHY, 2019)

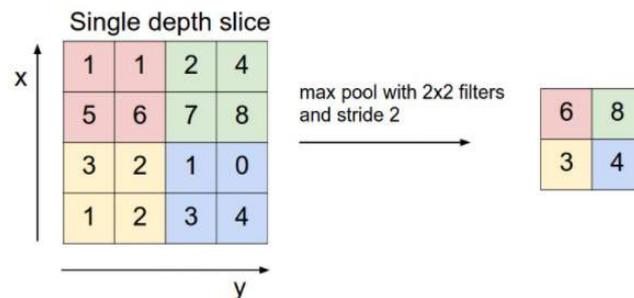
simplesmente selecionando o maior valor da sub-região, sendo esta chamada de Pooling de Máximo.

A Figura 3.4 demonstra de forma mais detalhada o funcionamento de uma Rede Neural Convolutiva básica, contendo os elementos mencionados anteriormente.

Figura 3.3 – Funcionamento de operações de Pooling em Redes Neurais Convolucionais.



(a) A partir de uma camada da Rede Neural Convolucional de entrada com dimensões de 224x224x64, uma operação de Pooling reduz para uma saída com volume de tamanho 112x112x64, preservando as características relevantes e preservando sua profundidade de filtros.



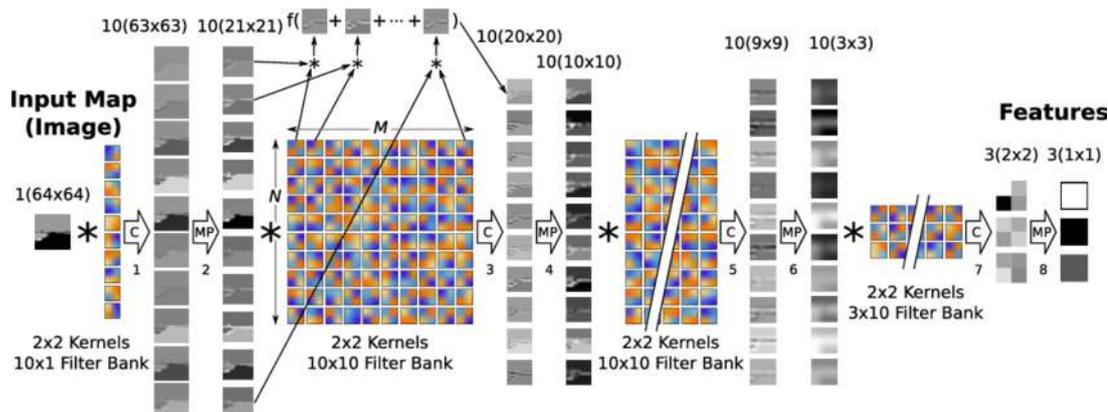
(b) A operação mais comum de Pooling é a operação de Máximo, onde o valor máximo de cada janela de entradas é tomado como o mais representativo para a sub-amostragem.

Fonte: (KARPATHY, 2019)

### 3.3.1 Treinamento de Redes Neurais Convolucionais

Por se tratar de uma classe de Rede Neural Artificial, e portanto uma técnica de aprendizado de máquina, Redes Neurais Convolucionais necessitam ser previamente treinadas. A técnica de aprendizado supervisionado mais comum é a *Backpropagation*, ou Retro-propagação. A retro-propagação consiste em computar o gradiente de uma função objetivo (também chamada de custo ou perda) para determinar como ajustar os pesos da Rede Neural, que são inicializados com valores aleatórios, de modo a minimizar os erros de classificação da rede e, para

Figura 3.4 – Ilustração do funcionamento de uma Rede Neural Convolutiva básica: O operador de convolução  $\otimes$  indica a convolução 2D de uma imagem ou mapa de características com um Banco de Filtros. O passo denotado por (C) diz respeito à convolução. O passo denotado por (MP) indica uma operação de Pooling de Máximo. O Classificador está sendo omitido da arquitetura, sendo representadas somente as operações que realizam a extração de características.



Fonte: (KOUTNÍK; SCHMIDHUBER; GOMEZ, 2014)

isto, é utilizado o algoritmo de Gradiente Descendente Estocástico (*Stochastic Gradient Descent*) (BOTTOU, 2010).

Para executar o treinamento da rede é necessário um conjunto de dados similares aos que se pretende posteriormente executar a classificação. Especificamente, para a detecção de objetos em imagens, é necessário obter um conjunto de imagens que contenham objetos das classes que se pretende detectar. Cada uma das imagens deve estar rotulada com as coordenadas das regiões de interesse em que cada objeto se encontra e a classe a qual este objeto pertence.

O número de imagens necessárias para realizar o treinamento varia de acordo com a arquitetura da rede, a complexidade da classe de objeto e outros parâmetros. O estado da arte indica que, em geral, as redes devem ser treinadas por conjuntos de milhares de imagens, o que torna esta a etapa mais custosa (KRIZHEVSKY; SUTSKEVER; HINTON, 2017).

A ausência de grandes conjuntos de dados rotulados de treinamento costuma ser um empecilho para o treinamento de Redes Neurais Convolucionais. Diante disso, foram disponibilizados publicamente, vários bancos de dados de imagens rotuladas como o Pascal VOC (*Visual Object Detection*) (EVERINGHAM et al., 2009) e, mais recentemente, o COCO (*Common Objects in Context*) (LIN et al., 2015). Esses bancos de dados contêm centenas de milhares de imagens rotuladas, com a presença de diferentes classes de objetos e se tornaram amplamente utilizados para o treinamento e ranqueamento de técnicas de detecção de objetos.

Para casos específicos, onde a classe de objeto a ser detectado não está presente nesses bancos de dados, e não há outro banco com um número expressivo de imagens rotuladas, uma alternativa é o uso de Aprendizagem por Transferência. Ao invés de iniciar o treinamento a partir de uma rede neural totalmente não treinada, a Aprendizagem por Transferência consiste em iniciar o treinamento em um modelo de rede neural já treinada para fins relacionados, ou seja, treinada para classes de objetos semelhantes às quais se pretende treinar. Esses modelos são usualmente chamados de modelos pré-treinados. Essa técnica permite obter modelos de redes neurais precisos, de forma mais rápida e com um conjunto reduzido de dados da classe a ser detectada (RAWAT; WANG, 2017). O Aprendizado por Transferência é possível pois as camadas menos profundas das redes neurais convolucionais tendem a representar características visuais mais elementares dos objetos, que são compartilhadas entre diferentes classes (RAZAVIAN et al., 2014).

### 3.3.2 Redes Neurais Convolucionais para Detecção de Objetos

Técnicas de Classificação de imagens, diferentemente das técnicas de Detecção de Objetos, apenas classificam uma imagem de entrada em uma classe de objeto, não indicando a região onde esse objeto está presente. A área de Classificação de Imagens se encontrava em um considerado platô desde o ano de 2010, pois desde então já não apresentava grandes avanços. Mas no ano de 2012, considerado como um marco para os avanços das técnicas de Classificação de Imagens, foi apresentada a rede neural convolucional conhecida como Alexnet, que combinou a capacidade de aprender um grande número de características visuais de objetos, inerente às ConvNets<sup>3</sup>, ao poder computacional das GPUs<sup>4</sup> da época. Os resultados foram expressivos, se tornando o estado da arte de desafios importantes, como o "IMAGENET Large Scale Visual Recognition Challenge" (ILSVRC2012) (KRIZHEVSKY; SUTSKEVER; HINTON, 2017).

Em 2016, foi proposto em (GIRSHICK et al., 2016) um método de Detecção de Objetos, nomeado R-CNN<sup>5</sup>, que consiste selecionar 2000 regiões da imagem na qual se deseja detectar objetos, por meio de um algoritmo de busca seletiva (UIJLINGS et al., 2013) e então alimentar cada uma destas regiões, que possivelmente contêm objetos, a uma Rede Neural Convolucional, como a AlexNet, ZFnet (ZEILER; FERGUS, 2013) ou VGG-16 (SIMONYAN; ZISSERMAN,

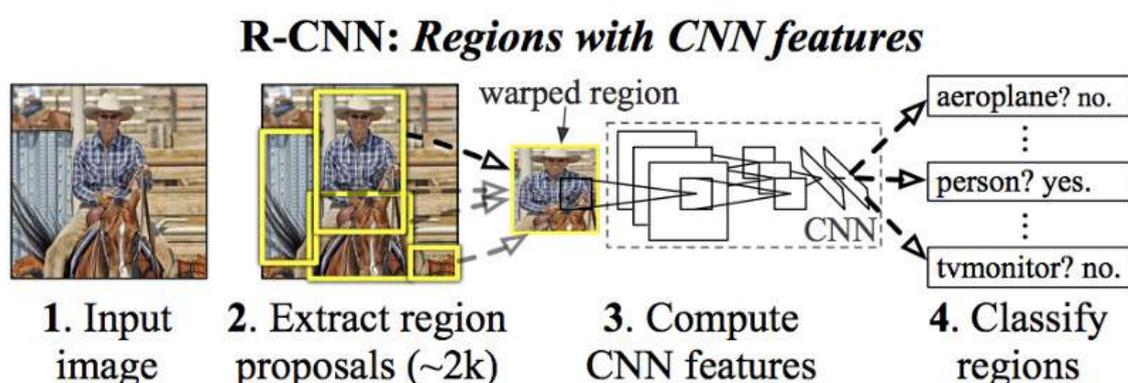
<sup>3</sup> ConvNet é um termo comum na literatura e se refere a Redes Neurais Convolucionais.

<sup>4</sup> O termo GPU se refere a Unidade de Processamento Gráfico

<sup>5</sup> O nome R-CNN se dá ao fato de se tratar em uma Rede Neural Convolucional (CNN) baseada em Regiões

2015), para a extração do mapa de características. As características extraídas pela ConvNet são então alimentadas em um classificador SVM linear, que estima a presença de alguma classe de objeto na região. A Figura 3.5 demonstra o funcionamento do detector R-CNN.

Figura 3.5 – Funcionamento do Detector de Objetos R-CNN: A partir de uma imagem de entrada (1) são extraídas 2000 propostas de região (2). Cada uma das regiões são alimentadas em uma ConvNet para extração de características (3) que são utilizadas para a classificação da região (4).



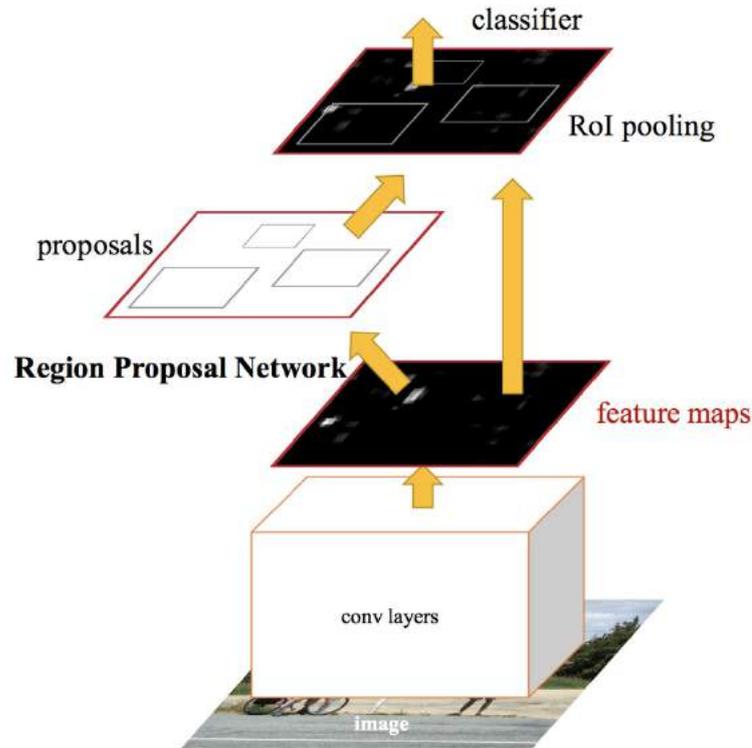
Fonte: (GIRSHICK et al., 2016)

Apesar de trazer uma precisão de detecção elevada frente às propostas da época (GIRSHICK et al., 2016), a estratégia de processar cada uma das 2000 regiões propostas consome um grande tempo, mesmo em GPU, levando em torno de 14 segundos por imagem. Para acelerar o processo de detecção de objetos, evitando processar um grande número de regiões propostas, foi elaborada a técnica conhecida como Faster-R-CNN (ALAMSYAH; FACHRURROZI, 2019), que elimina o algoritmo de busca seletiva e alimenta a imagem inteira em uma rede convolucional para a proposição de regiões (RPN - Region Proposal Network) que tem como saída um conjunto de regiões retangulares, cada uma com uma pontuação em relação a conter ou não um objeto. A Figura 3.6 ilustra a topologia de funcionamento do detector Faster-RCNN.

O funcionamento da rede RPN (Region Proposal Network) está representado na Figura 3.7, e funciona da seguinte maneira:

- Uma janela de tamanho  $n \times n$  (tipicamente,  $n = 3$ ) percorre o mapa de características da última camada da ConvNet de extração de características;
- A cada posicionamento desta janela no mapa de características são geradas  $k$  propostas de regiões, chamadas de âncoras, centralizadas na janela em questão. Cada âncora possui

Figura 3.6 – Funcionamento do Detector de Objetos Faster-R-CNN: A imagem de entrada é alimentada à Rede Neural Convolutiva de extração de características. O Mapa de Características é então utilizado como entrada para a Rede RPN que identifica Propostas de Regiões de objetos. Essas regiões são então redimensionadas por meio de uma camada de pooling de região de interesse (RoI). Por fim, essas regiões são classificadas em classes de objetos.



Fonte: (ALAMSYAH; FACHRURROZI, 2019)

uma variação única de escala (altura, largura e/ou proporção) de sua janela, chamada caixa de âncora.

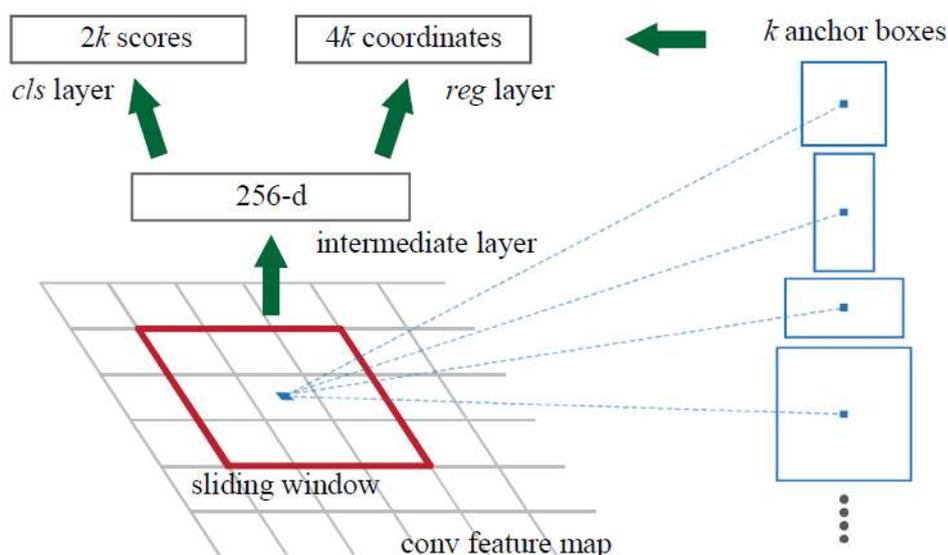
- Cada caixa de âncora recebe uma pontuação em relação a conter ou não um objeto, e aquelas com as maiores pontuações são consideradas propostas de região.

Para limitar o esforço computacional, um número máximo de propostas de regiões é estabelecido.

### 3.3.3 Arquitetura Inception de Rede Neurais Convolucionais

A forma mais simples de melhorar o desempenho de uma Rede Neural Convolutiva é aumentando seu tamanho. Isto inclui aumentar sua profundidade, ou seja, seu número de camadas, e também aumentar a largura de cada camada (SZEGEDY et al., 2015a). Mas com o aumento no tamanho das redes neurais artificiais, há um aumento no seu número de parâmetros, o que trás duas grandes desvantagens:

Figura 3.7 – Ilustração do funcionamento da Region Proposal Network: a partir de cada posicionamento de uma janela que percorre o mapa de características, são geradas  $k$  caixas âncora, definidas por suas coordenadas e pontuações, que podem então ser selecionadas como propostas de região.



Fonte: (ALAMSYAH; FACHRURROZI, 2019)

- Redes neurais com um número grande de parâmetros são suscetíveis a um sobreajuste, principalmente quando o conjunto de dados de treinamento é limitado, reduzindo sua capacidade de generalizar;
- Um grande número de parâmetros exige um hardware capaz de suportá-los, limitando o número de casos em que o uso da rede neural se torna viável.

Sendo assim, arquiteturas de redes neurais artificiais que conseguem melhorar o desempenho com estratégias de otimização são de grande relevância.

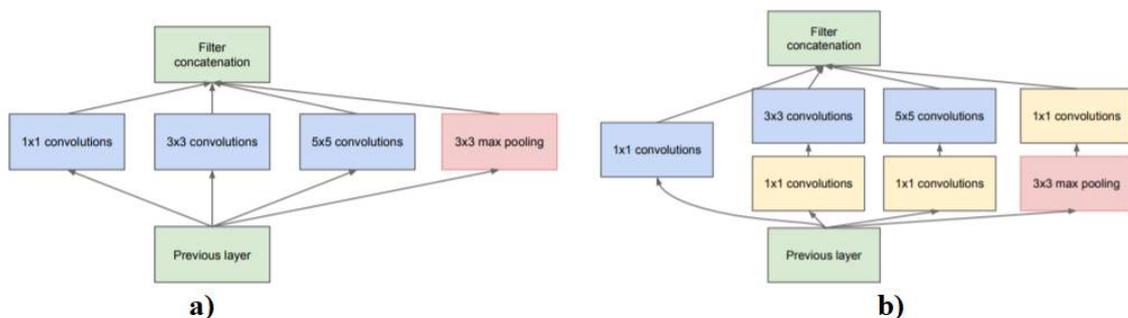
### 3.3.3.1 Inception versão 1

A Arquitetura *Inception*, proposta inicialmente por Szegedy et al. (2015a), e conhecida como *Inception v1* traz como proposta, a melhoria de desempenho de ConvNets por meio do aumento da largura das camadas de convolução, realizando operações de convolução de diferentes tamanhos em uma mesma camada (usualmente convoluções de tamanho  $3 \times 3$  ou  $5 \times 5$ ), e então concatenando os resultados, como representado na Figura 3.8a. Essa estratégia, conhecida como *módulo Inception* permite melhorar o desempenho da rede, pois convoluções com tamanhos menores são melhores em encontrar características distribuídas de forma mais local,

enquanto convoluções maiores são melhores em encontrar características globalmente distribuídas.

Por aumentar o número de operações de convolução, módulos Inception aumentam o número de parâmetros da rede. Por esse motivo, em (SZEGEDY et al., 2015a) é proposto realizar uma operação de convolução de tamanho  $1 \times 1$  com um número reduzido de filtros antes que operações de convolução de diferentes tamanhos sejam realizadas. A Figura 3.8b representa este tipo de estratégia, que reduz o esforço computacional e o número de parâmetros, mas preserva as vantagens proporcionadas pelo módulo Inception.

Figura 3.8 – a) Módulo *Inception* sem redução de filtros e b) Módulo Inception com redução de filtros



Fonte: (SZEGEDY et al., 2015a)

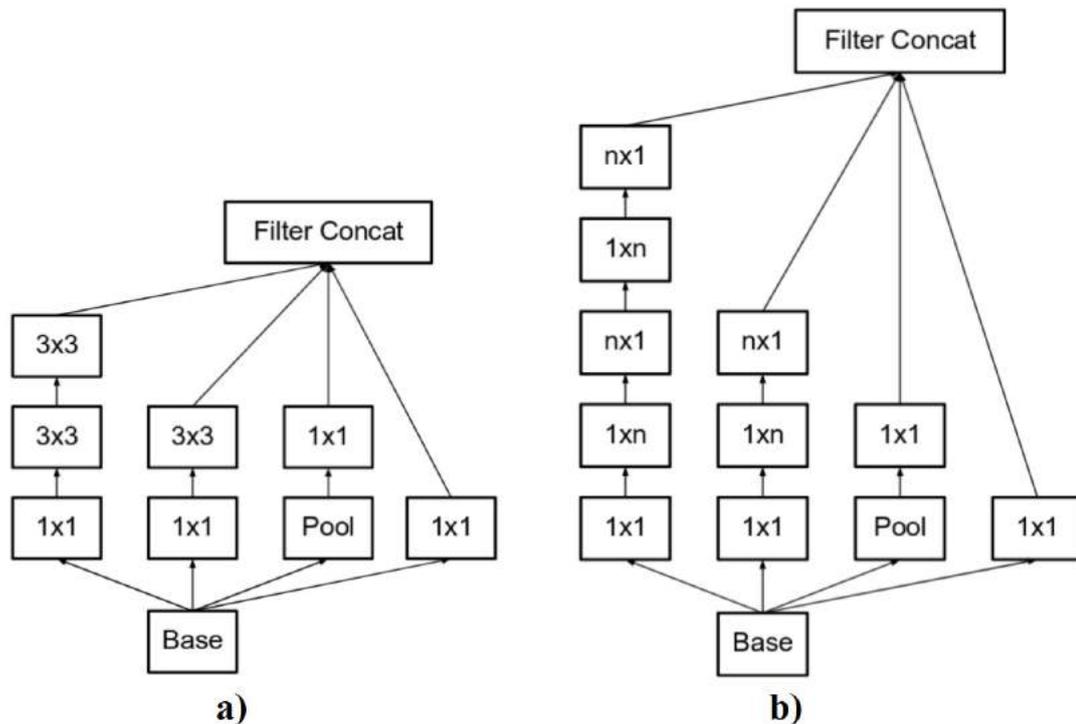
### 3.3.3.2 Inception Versão 2

Ainda em 2015, uma série de melhorias para módulos *Inception* foram propostas em (SZEGEDY et al., 2015b), conhecidas como *Inception V2*. Essas melhorias buscaram aumentar a precisão da rede neural convolucional e reduzir a complexidade computacional de suas convoluções.

Um exemplo dessas melhorias é a fatoração de operações de convolução de tamanho  $5 \times 5$  em duas convoluções de tamanho  $3 \times 3$ , de modo a reduzir os custos computacionais. Uma convolução de tamanho  $5 \times 5$  é 2,78 vezes mais dispendiosa computacionalmente do que uma convolução  $3 \times 3$ , justificando a melhoria de custo computacional. Essa fatoração é ilustrada na Figura 3.9a.

Outro exemplo de otimização é a fatoração de uma operação de convolução de filtro  $n \times n$  em uma combinação de operações  $1 \times n$  e  $n \times 1$ , como representado na Figura 3.9b. Apesar de serem operações equivalentes, a fatoração traz uma redução de 33% no custo computacional.

Figura 3.9 – Exemplos de estratégias de Módulos Inception Versão 2. a) Fatoração  $5 \times 5$  em duas  $3 \times 3$  e b) Fatoração  $1 \times n$  e  $n \times 1$



Fonte: (SZEGEDY et al., 2015a)

### 3.4 Desempenho de detectores de objetos

Para avaliar o desempenho de um classificador de imagens em um determinado problema, este deve ser testado em conjunto de imagens de teste que o resultado esperado de classificação das imagens é previamente conhecido. Após a execução do classificador, os resultados podem ser agrupados da seguinte forma (FAWCETT, 2006):

- Verdadeiro Positivo ( $vp$ ): resultado esperado positivo e classificação positiva,
- Verdadeiro Negativo ( $vn$ ): resultado esperado negativo e classificação negativa,
- Falso Positivo ( $fp$ ): resultado esperado negativo e classificação positiva,
- Falso Negativo ( $fn$ ): resultado esperado positivo e classificação negativa.

Com estes resultados, a precisão ( $P$ ) do classificador é definida como:

$$P = \frac{vp}{vp + fp} \quad (3.2)$$

e a revocação ( $R$ ) do classificador é definida como:

$$R = \frac{vp}{vp + fn}. \quad (3.3)$$

Para um detector de objetos, a tarefa é localizar, por meio de regiões de interesse, cada instância de objeto presente na imagem e então classificar cada uma delas. Portanto, para que um resultado seja definido, por exemplo, como Verdadeiro Positivo ( $vp$ ), é necessário avaliar se:

1. A localização da região de interesse resultante coincide com a região esperada;
2. O objeto contido na região foi classificado corretamente de acordo com a classe esperada.

Para isto, é amplamente aceito na literatura o conceito de Intersecção sobre União ( $IoU$ ) (EVERINGHAM et al., 2009):

$$IoU = \frac{area(R_r \cap R_e)}{area(R_r \cup R_e)}, \quad (3.4)$$

onde  $R_r$  é a região de interesse resultante do detector e  $R_e$  a região de interesse esperada para a instância. Considerando um limiar para o valor de  $IoU$  (0,5, por exemplo), cada instância de detecção será agrupada de acordo com os seguintes critérios:

- Verdadeiro Positivo ( $vp$ ): se  $IoU$  for superior ou igual ao limiar,
- Falso Positivo ( $fp$ ): se  $IoU$  for inferior ao limiar,
- Falso Negativo ( $fn$ ): caso nenhuma região de interesse seja definida para a instância de objeto.

Valores de Verdadeiro Negativo  $vn$  não são contabilizados, visto que quaisquer regiões onde não estão presentes instâncias de objetos e não há nenhuma falsa detecção, podem ser consideradas Verdadeiro Negativo  $vn$ .

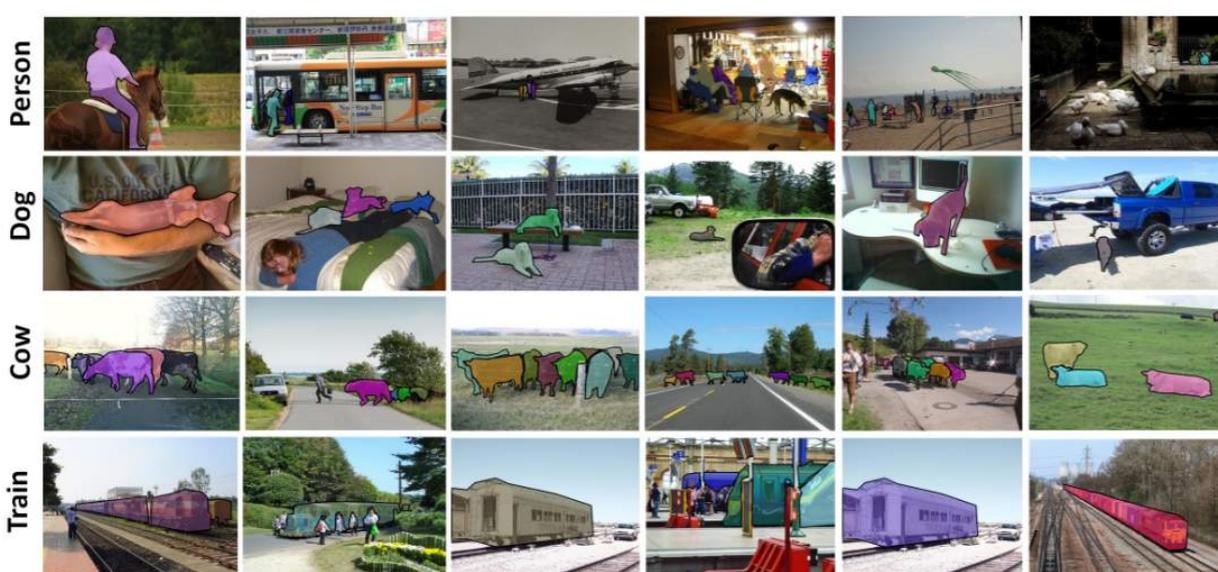
## 4 METODOLOGIA

Neste capítulo será apresentada a metodologia tomada para executar a solução de detecção e rastreamento visual de suínos em granjas, bem como o registro do comportamento alimentar de cada animal individualmente, informação relevante para a alimentação de precisão. Para isso, inicialmente será apresentada a metodologia tomada para o treinamento da rede neural convolucional utilizada para a detecção dos animais na imagem, e em seguida será apresentado o diagrama geral da solução de detecção e rastreamento, bem como cada um de seus passos.

### 4.1 Treinamento da Rede Neural Convolucional para Detecção de Suínos

Para o treinamento e execução da rede neural convolucional, foi utilizado o framework *Tensorflow*, que possui uma implementação de um modelo de Rede Neural Convolucional tipo Faster-R-CNN, com módulos Inception Versão 2. Essa implementação está de acordo com a arquitetura descrita em (SZEGEDY et al., 2015b), e o modelo utilizado<sup>1</sup> se encontrava previamente treinado com imagens do banco de dados COCO (LIN et al., 2015), que possui mais de 200 mil imagens de 80 categorias diferentes de objetos. A Figura 4.1 contém exemplos de imagens presentes no banco de dados COCO.

Figura 4.1 – Exemplo de imagens rotuladas presentes no banco de dados COCO.



Fonte: (LIN et al., 2015)

<sup>1</sup> O modelo utilizado pode ser encontrado no seguinte repositório: <https://github.com/tensorflow/models>.

É uma boa prática para o treinamento de redes neurais convolucionais para detecção de objetos, o uso de imagens em cenários diversificados e com o objeto em diferentes escalas e posições, uma vez que proporciona redes treinadas menos enviesadas e mais robustas às mudanças de cenário. Sendo assim, em busca de imagens rotuladas que contenham suínos em cenários diversos, foram investigados alguns dos grandes bancos de dados de imagens rotuladas, como o ImageNet, COCO<sup>2</sup> e OpenImages da Google. Porém, os dois primeiros não dispunham da classe de objeto buscada, e no último tinha poucas imagens de boa qualidade, ou em ângulos e escalas semelhantes às imagens que seriam alimentadas à rede neural treinada.

Dada a dificuldade em encontrar imagens rotuladas de suínos em granjas, foram extraídos 400 quadros aleatórios de vídeos de câmeras de monitoramento de um experimento de preferência alimentar com suínos em fase de crescimento, realizado na Universidade Federal de Lavras em seu Departamento de Zootecnia. Todas as imagens obtidas possuem uma resolução de 1280 por 720 pixels. A Figura 4.2 é um exemplo de imagem utilizada para o treinamento.

Figura 4.2 – Exemplo de imagem utilizada para o treinamento e teste da rede neural convolucional.



Fonte: do autor.

As imagens coletadas foram então rotuladas manualmente utilizando a ferramenta de código aberto *LabelImg*<sup>3</sup>, sendo 80% delas destinadas ao conjunto de treinamento e 20% para o conjunto de validação da rede neural convolucional. Mais detalhes sobre o treinamento serão mostrados no Capítulo 5.

Para realizar a etapa de treinamento, os seguintes parâmetros foram estabelecidos:

<sup>2</sup> Do inglês "*Common Objects in Context*".

<sup>3</sup> Disponível em: <https://github.com/tzutalin/labelImg>.

- Função de perda: Softmax,
- Tamanho de Batch<sup>4</sup>: 1,
- Taxa de Aprendizado<sup>5</sup>: 0,0002,
- Número máximo de propostas da RPN: 300.

Esses parâmetros se referem ao algoritmo de *Backpropagation* mencionado no Capítulo 2, utilizado para o treinamento da rede neural convolucional. O treinamento foi realizado em um computador com as seguintes configurações de hardware e software:

- Processador: Intel Core i5 9400F,
- Placa de Vídeo: Nvidia RTX 2060 6GB,
- Memória RAM: 16GB DDR4,
- Sistema Operacional: Windows 10,
- Linguagem de Programação: Python 3.5.6<sup>6</sup>,
- Framework: Tensorflow GPU 1.10<sup>7</sup>.

## 4.2 Diagrama geral da solução de detecção e rastreamento

A solução apresentada neste trabalho pode ser dividida em quatro passos, descritos no diagrama da Figura 4.3. Essas etapas são:

1. Aquisição da Imagem;
2. Detecção via Rede Convolucional;
3. Rotulação dos indivíduos;
4. Registro de comportamento alimentar.

---

<sup>4</sup> O *tamanho de batch* corresponde ao número de imagens do conjunto de treinamento que serão utilizadas em cada etapa de retro-propagação.

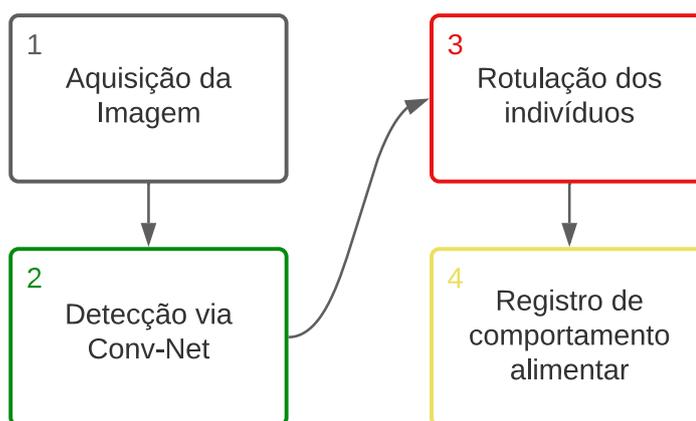
<sup>5</sup> A *taxa de aprendizado* é o fator de ajuste dos pesos da rede neural convolucional a cada passo de treinamento.

<sup>6</sup> Disponível em: <https://www.python.org/downloads/release/python-356/>

<sup>7</sup> Disponível em: <https://www.tensorflow.org/install/pip>

Esse diagrama será melhor detalhado nas seções que seguem. O código-fonte da metodologia do trabalho está publicamente disponível em <<https://github.com/Geraldo-Neto/CNN-Pig-Detection-and-Tracking>>.

Figura 4.3 – Diagrama da solução de detecção de rastreamento de suínos, incluindo registro de comportamento alimentar. (1) Aquisição da imagem de entrada, (2) Detecção dos animais via Rede Convolutiva Faster-R-CNN, (3) Rotulação dos indivíduos e (4) Registro de tempo e frequência alimentar.



Fonte: do autor.

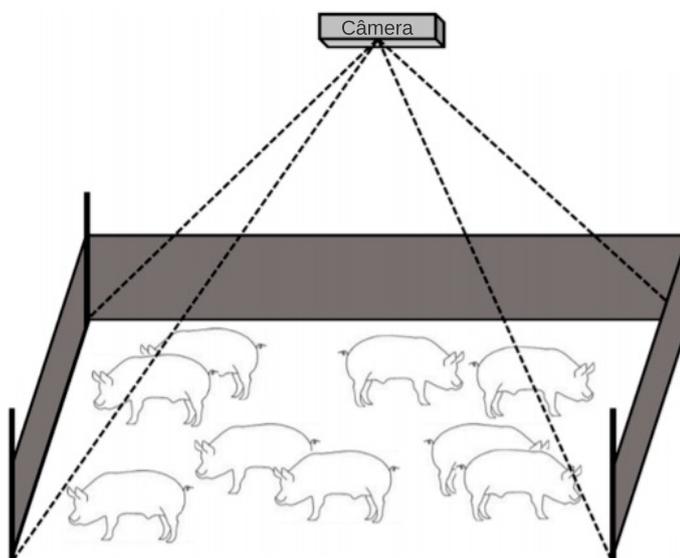
### 4.3 Aquisição da Imagem

O cenário considerado para o rastreamento é aquele onde o posicionamento da câmera é feito de modo a obter uma vista de topo dos animais. Sendo assim, a câmera deve estar perpendicular ao plano da baía dos animais, ou o mais próximo disso. Essa suposição permite reduzir a complexidade do algoritmo, pois permite tratar o problema como um rastreamento 2D, no plano da imagem, como ilustrado na Figura 4.4. Além disso, este posicionamento de câmera minimiza situações onde um animal possa ocluir outro animal.

Como a detecção e rastreamento em vídeo é realizada quadro a quadro, o primeiro passo a ser tomado é o de aquisição da imagem. Nesse passo é adquirido um quadro do vídeo a ser processado. Caso o vídeo a ser processado possua um número elevado de quadros por segundo, a amostragem pode ser feita de modo intermitente, capturando um quadro a cada N quadros disponíveis na sequência do vídeo.

A resolução do quadro a ser processado é fator de grande impacto no desempenho de processamento e, portanto, neste passo também pode ser feito um redimensionamento do quadro. O redimensionamento é necessário em situações onde o hardware que executará o processamento tem desempenho reduzido, ou quando se deseja um processamento mais rápido.

Figura 4.4 – Ilustração do posicionamento ideal de câmera para o rastreamento dos animais.



Fonte: (SA et al., 2019).

Resoluções muito baixas, no entanto, resultam em detecções mais imprecisas. Em geral, nos experimentos realizados, resoluções acima de 640 por 360 pixel são suficientes para se obter bons resultados em termos de acurácia.

As imagens do experimento em que a solução se apoiou para realizar os testes não foram capturadas na situação ideal de posicionamento de câmera ilustrada na Figura 4.4, uma vez que as câmeras não estavam perpendiculares ao plano das baias. Isto se deve ao fato de que, inicialmente, o experimento realizado não visava a aplicação de técnicas de visão computacional, apenas a observação humana e anotação manual de tempo de alimentação dos animais e, portanto, não se atentou ao posicionamento ideal de câmeras para este tipo de aplicação. Por esse motivo, as baias consideradas para o rastreamento são somente aquelas mais próximas da câmera, evitando grandes distorções. A Figura 4.2 ilustra uma imagem adquirida pelo sistema.

#### 4.4 Detecção dos animais via Rede Neural Convolutional

Seguindo a Figura 4.3, o segundo passo é a detecção dos animais presentes na imagem. Esse processo é feito por meio da alimentação da imagem como entrada para a Rede Neural Convolutional Faster-R-CNN, apresentada na Seção 3.3.2 que foi então treinada para executar a detecção de suínos.

A saída da rede neural convolutacional mediante a entrada de uma imagem, é um conjunto de regiões de interesse retangulares, que indicam a presença do objeto e suas respectivas

pontuações de detecção. A pontuação de uma certa detecção denota a porcentagem de certeza dessa detecção. A fim de evitar falsos positivos, detecções com uma pontuação abaixo de um valor limiar são desprezadas, o resultado deste passo está exemplificado na Figura 4.6b.

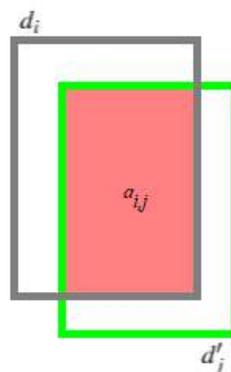
#### 4.5 Identificação dos Indivíduos

Realizadas as detecções dos animais, como ilustrado na Figura 4.3, o próximo passo é rotulá-los com um número de identificação, a fim de que o rastreamento de cada animal seja feito ao longo de toda a sequência de imagens do vídeo.

Para a imagem inicial, a rotulação é feita de modo arbitrário, atribuindo aleatoriamente um número de identificação único a cada animal detectado na imagem atual  $d'_j$ . Para as demais imagens, a rotulação é feita comparando cada detecção da imagem anterior  $d_i$  com cada detecção na imagem atual  $d'_j$ , de modo a definir as melhores correspondências.

Considerando que o deslocamento de cada animal entre duas imagens sequenciais é de uma distância limitada<sup>8</sup>, detecções correspondentes<sup>9</sup> da imagem anterior  $d_i$  e da imagem atual  $d'_j$  devem possuir uma área de intersecção  $a_{i,j}$ , como esquematizado na Figura 4.5. Caso alguma detecção atual se interseque com mais de uma detecção anterior, será considerada válida aquela que possui menor distância euclidiana  $\delta_{i,j}$  entre seu centro  $c_i$  e o centro da detecção atual  $c'_j$ . Caso não haja uma intersecção válida entre um elemento da imagem atual e da passada, para uma detecção atual, à detecção na imagem atual será atribuído um novo número de identificação, não presente nos quadros anteriores.

Figura 4.5 – Área de intersecção entre detecções de quadros consecutivos.



Fonte: do autor.

<sup>8</sup> Nos experimentos realizados, vídeos com taxas de quadros por segundo maiores que 15 são suficientes para que esta suposição seja válida.

<sup>9</sup> Detecções correspondentes são detecções em diferentes imagens que se referem ao mesmo animal.

Portanto, sendo  $ID$  um vetor de rotulações, a identificação de uma detecção atual  $d'_j$  será:

$$ID(d'_j) = \begin{cases} ID(d_i) & \text{se } a_{i,j} > 0, \\ ID(\min(\delta_{i,j})) & \text{se houver mais de uma intersecção válida para } d_i, \text{ ou} \\ \max(ID) + 1 & \text{caso contrário.} \end{cases} \quad (4.1)$$

#### 4.6 Registro de Comportamento Alimentar

Após a rotulação e identificação de cada animal, a última etapa é o registro de alimentação dos indivíduos. Para se alimentarem, os animais devem visitar os cochos de alimentação, que são áreas fixas, pré-identificadas na imagens, que contêm o alimento, como indicado com sombreamento verde na Figura 4.6b. Para verificar a alimentação, o algoritmo efetua o cálculo de áreas de intersecção entre o animal rotulado e as regiões de alimentação. Intersecções com área maior que um limiar definido são assumidas como positivo para o registro alimentar do individuo na imagem atual. Para fins de demonstração, animais em alimentação são anotados com a palavra "*Feeding*", como ilustrado na Figura 4.6 item (b) e (c). Assim, pode ser contabilizado o tempo,  $Ta(x)$ , em que cada indivíduo  $x$  permaneceu em alimentação, bem como o número  $Na(x)$  de vezes que ele visitou a região do cocho de alimentação.

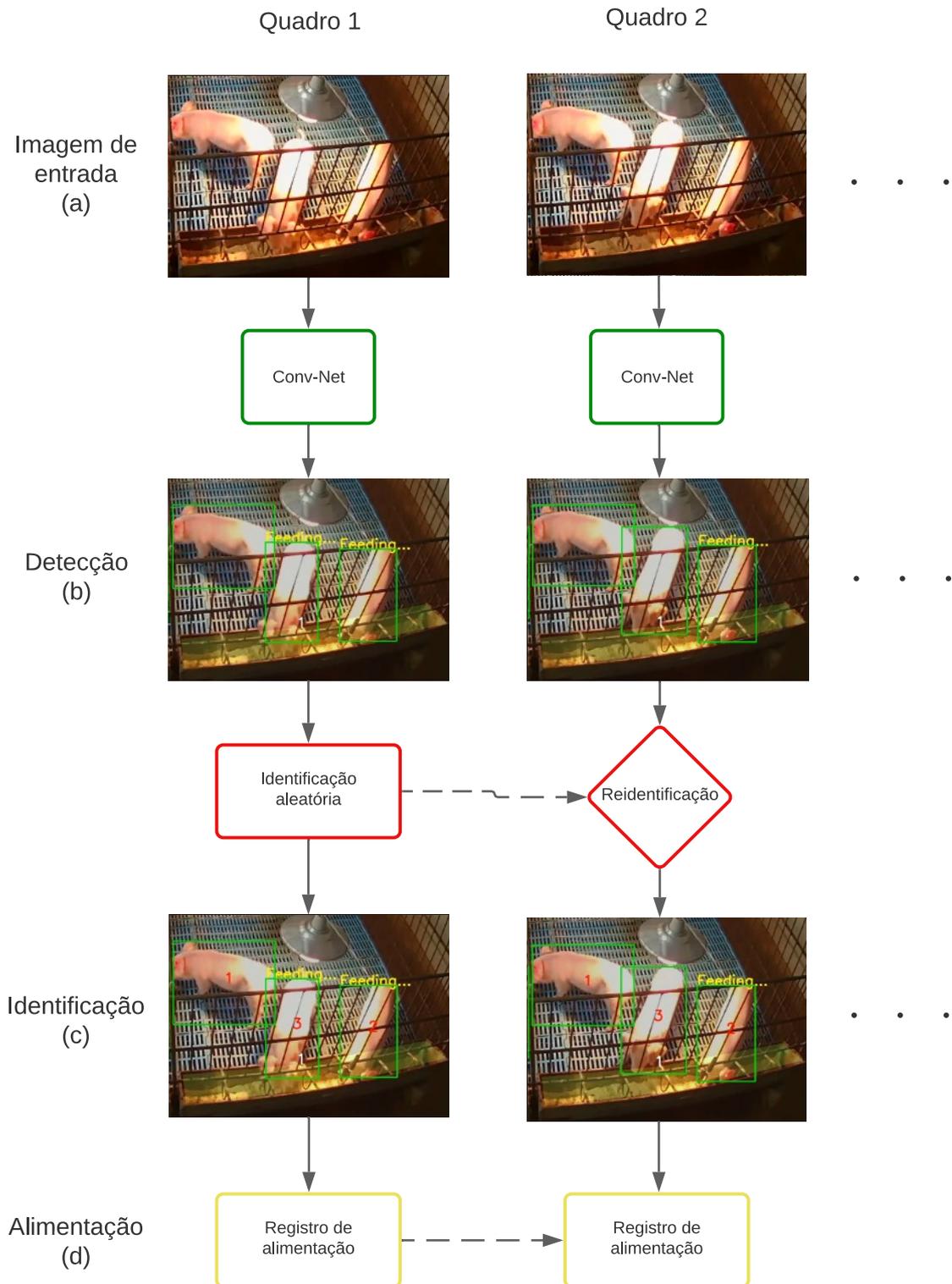
Sabendo-se que o vídeo possui uma taxa de quadros por segundos constante  $FPS$  (do inglês *frames per second*), e que o indivíduo foi considerado como "*alimentando*" por  $n$  quadros, o tempo de alimentação é dado por:

$$Ta(x) = n(x)/FPS \quad (4.2)$$

A Figura 4.6 é o diagrama geral da solução, demonstrando como é feito o processo, tomando como exemplo dois quadros sequenciais. A região sombreada em verde corresponde a região em que está presente o cocho de alimentação.

No próximo capítulo, a metodologia aqui apresentada será aplicada em dados reais para a análise da eficácia da solução.

Figura 4.6 – Diagrama da solução de detecção de rastreamento de suínos com a exemplificação de um caso real com dois quadros consecutivos. (a) Amostragem de uma imagem de entrada, (b) Resultado da detecção via Rede Convolutiva, (c) Resultado da identificação de cada indivíduo e (d) Registro de tempo e frequência alimentar.



Fonte: do autor.

## 5 RESULTADOS

Neste capítulo serão apresentados e analisados os resultados qualitativos e quantitativos da proposta abordada no Capítulo 4. Primeiramente, na Seção 5.1, serão expostas as configurações de treinamento, hardware e software utilizados para obter o modelo treinado da ConvNet utilizada para a detecção dos animais. A Seção 5.2 demonstra a Interface Homem-máquina desenvolvida para facilitar a utilização do sistema. Em seguida, nas Seções 5.3 e 5.4, respectivamente, serão abordados os resultados qualitativos e quantitativos do treinamento da ConvNet e do sistema de rastreamento e registro de comportamento alimentar como um todo.

É importante ressaltar que este trabalho define seu escopo de forma específica: registrar o comportamento alimentar de suínos em ambientes de granjas, por meio do rastreamento visual com câmeras monoculares e com vista superior dos animais. Portanto, as discussões sobre os resultados serão feitas dentro dos limites desse escopo. Além de poder ser utilizado como forma de implementar a alimentação de precisão na indústria, o sistema proposto pode também ter relevância em experimentos de preferência alimentar, que são de grande importância tanto na academia quanto na indústria de nutrição animal.

Esses experimentos podem ser realizados em um cenário com animais isolados, onde está presente somente um animal por baia, quanto de forma coletiva, com múltiplos animais por baia. Os resultados serão portanto discutidos visando ambos os cenários, sendo que o primeiro será chamado de Cenário 1 e o segundo será chamado de Cenário 2, conforme indica a Figura 5.1.

### 5.1 Treinamento da Rede Neural Convolutacional para Detecção de Suínos

O treinamento da Rede Neural Convolutacional para a detecção de Suínos, durou cerca de 3 horas, atingindo cerca de 70 mil etapas de treinamento da rede. Para avaliar o desempenho da rede após o treinamento, foi levantado um conjunto de 25 imagens de teste, que não estavam presentes nos conjuntos de treinamento e validação, que continham múltiplos animais em granja. O resultado está demonstrado na Tabela 5.1.

De acordo com os dados da Tabela 5.1, foi calculada a precisão e revocação do detector para o conjunto de imagens de teste. Os resultados estão presentes na Tabela 5.2 e demonstram que a detecção tem uma precisão superior a 95%, porém uma revocação abaixo dos 90%, o que se deve ao número de animais que estavam com grande parte do corpo sob oclusão e não foram detectados, contabilizando falsos negativos.

Figura 5.1 – Cenários considerados para os testes.



(a) Cenário 1, onde há apenas um animal por baia.



(b) Cenário 2, onde há mais de um animal por baia.

Fonte: do autor.

## 5.2 Interface de Usuário

Para que a contabilização de tempo de alimentação dos animais seja realizada pelo sistema, é necessário selecionar o arquivo de vídeo a ser processado, além de selecionar as regiões da imagem em que estão posicionados os comedouros. Com o objetivo de facilitar esta seleção, para se tornar viável o uso do sistema em diferentes granjas e em diferentes posicionamentos de câmera, foi desenvolvida uma interface de usuário para o sistema. Esta interface, desenvolvida na linguagem de programação Python 3.5, é apresentada na Figura 5.2(a). Por meio desta inter-

Tabela 5.1 – Tabela verdade de detecção de animais para o conjunto de teste.

Verdadeiro Positivo	Falso Positivo	Falso Negativo
93 (85%)	4 (3%)	12 (11%)

Fonte: do autor.

Tabela 5.2 – Precisão de Revocação do detector de objetos treinado.

Precisão	Revocação
95,8%	88,5%

Fonte: do autor.

face, é possível selecionar o arquivo de vídeo a ser processado, o trecho do vídeo desejado, o número de comedouros e suas regiões, representadas por polígonos, como demonstrado na Figura 5.2(b). Ao final do processamento do vídeo, o sistema gera uma tabela em formato Excel com os resultados de tempo de alimentação de cada animal.

Figura 5.2 – Interface de Usuário desenvolvida para o sistema.



(a) Interface de Usuário Principal.



(b) Seleção de regiões de alimentação.

Fonte: do autor.

### 5.3 Resultados Qualitativos

A Figura 5.3(a) demonstra resultados da detecção da rede convolucional dos animais no Cenário 1. Neste cenário, onde a oclusão entre os animais é mínima, tanto a detecção quanto o rastreamento se tornam menos desafiadores para o algoritmo proposto, sendo este capaz de atingir os objetivos propostos, como será apresentado na Seção 5.4. No experimento realizado, quando o ambiente se tornava escuro, as câmeras ligavam automaticamente sua função noturna, capturando imagens em escala de cinza, algo que não foi previsto no treinamento da ConvNet. Contudo, apesar do treinamento não ter tomado atenção à tais ambientes, o detector conseguiu realizar a detecção dos animais com boa precisão, como demonstrado na Figura 5.3(b).

Considerando o Cenário 2, onde os animais estão em conjunto nas baias, o detector conseguiu bom desempenho, mesmo com oclusões parciais entre animais e entre animal e ambiente, como demonstrado na Figura 5.3(c). Esse resultado se repete, mesmo em granjas com um grande número de animais, como demonstrado na Figura 5.3(d). No entanto, o comportamento natural do animal em alguns momentos pode dificultar muito a tarefa de detecção e principalmente, a tarefa de rastreamento.

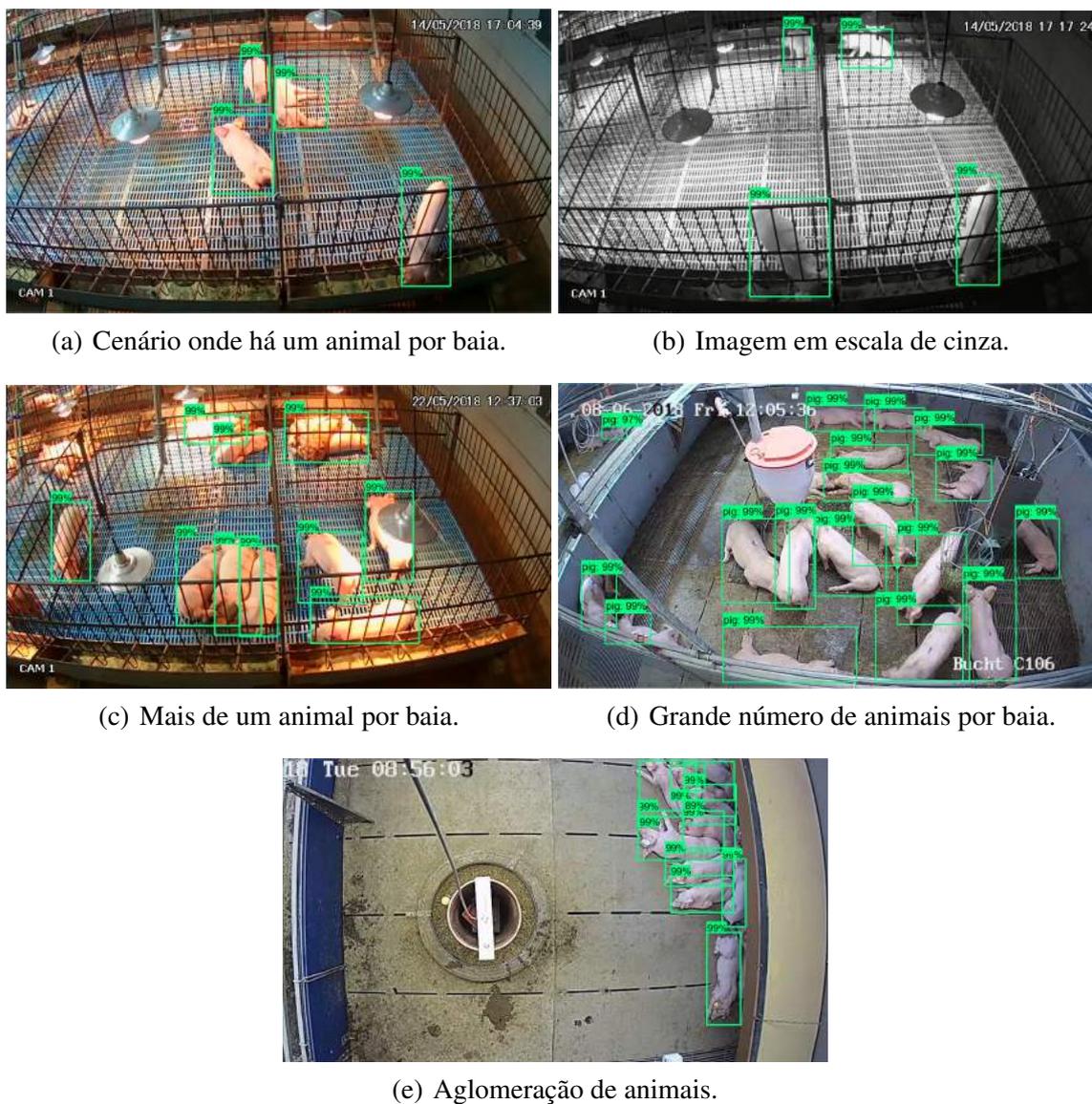
Em ambientes mais frios, os animais tendem a se aglomerar, resultando em muitas oclusões parciais e até mesmo oclusões totais, como demonstrado na Figura 5.3(e). Nessas situações, é notável a dificuldade em detectar e rastrear individualmente cada animal, mesmo para um ser humano.

Em ambientes onde existem outros animais além de suínos, que são ambientes diferentes daquele inicialmente proposto, o número de falsos positivos é grande. Isso demonstra uma deficiência de diversificação do banco de dados utilizado para o treinamento. Essa deficiência, no entanto, não impacta o objetivo proposto, visto que este se limita a ambientes de granjas de suínos.

### 5.4 Resultados Quantitativos

Para a configuração de hardware e software utilizado, o tempo de processamento de detecção para cada imagem com resolução de 1280x720 foi em média 77 milissegundos, resultando em uma taxa de aproximadamente 13 quadros por segundo. Como descrito no Capítulo 4, a detecção pode ser feita de modo intermitente. Portanto, nos experimentos a detecção foi rea-

Figura 5.3 – Resultados do detector em diferentes cenários.



Fonte: do autor.

lizada uma vez a cada dois quadros, resultando em uma taxa de quadros de aproximadamente 25 quadros por segundo.

A fim de avaliar o resultado do sistema como um todo, foi retirado para cada cenário (1 e 2) um trecho aleatório de vídeo com 10 minutos de duração de uma câmera de monitoramento do experimento. Neste trecho, foi comparado o tempo de alimentação contabilizado pelo sistema com aquele manualmente contabilizado. As baias consideradas para a avaliação foram aquelas mais próximas à câmera, devido a menor distorção e menores chances de oclusões causadas pelo ambiente. Os resultados para o Cenário 1 se encontram na Tabela 5.3.

Ao realizar os experimentos para o Cenário 2, foi constatado um grande número perdas de rastreamento, devido às interações entre os animais que ocasionam oclusões parciais ou

Tabela 5.3 – Tabela de tempo de alimentação (em segundos) para o Cenário 1.

Comedouro	Animal 1	Animal 2	Animal 1 (esperado)	Animal 2 (esperado)
1	561 s	0	552 s	0
2	0	477 s	0	465 s

Fonte: do autor.

totais, como aquelas citadas no início deste capítulo e demonstradas nas Figuras 5.3(c) e 5.3(e). Estas perdas de rastreamento resultaram em informações individuais muito imprecisas. Face a isto, para o Cenário 2 não foram considerados os tempos de alimentação para cada indivíduo, mas a soma de seus tempos de alimentação em cada comedouro. Os resultados são mostrados na Tabela 5.4

Tabela 5.4 – Tabela de tempo de alimentação (em segundos) para o Cenário 2.

Comedouro	Baia 1	Baia 2	Baia 1 (esperado)	Baia 2 (esperado)
1	662 s	0	647 s	0
2	0	0	0	0

Fonte: do autor.

É importante ressaltar que no Cenário 2 haviam animais presentes na Baia 2 e estes se movimentavam no trecho de vídeo analisado, mas de fato não se alimentaram em nenhum momento.

Nos experimentos realizados, foi possível notar que há um pequeno atraso entre o momento em que o animal ocupa a área de alimentação e o momento em que ele de fato inicia sua alimentação. O mesmo ocorre no momento em que o animal interrompe sua alimentação. Como o sistema assume que o animal está se alimentando ao ocupar a área de alimentação, o tempo contabilizado pelo sistema pode ser maior do que o tempo real de alimentação. Essas diferenças estão de acordo com as Tabelas 5.3 e 5.4.

## 6 CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi propor um sistema de detecção e rastreamento visual 2D de suínos em granjas que possa ter aplicação tanto na alimentação de precisão, quanto na experimentação alimentar na suinocultura. Mesmo com um pequeno conjunto de 400 imagens para treinamento e validação, a Rede Neural Convolutiva treinada para a detecção dos animais obteve bons resultados, sendo capaz de detectar os animais em diferentes posições, iluminações, e até mesmo sob oclusões parciais.

A estratégia proposta de detectar animais que estão se alimentando e contabilizar o tempo de alimentação destes, mostrou resultados próximos ao esperado, como elucidado no Capítulo 5. Porém, devido ao comportamento de intensa interação entre os animais em granjas abertas, situações de oclusões parciais ou completas entre os animais são muito recorrentes. Essas oclusões elevam drasticamente a complexidade de rastreamento exigida do algoritmo. Portanto, o algoritmo de rastreamento proposto, apesar de ser capaz de rastrear o movimento de um indivíduo isolado, não foi capaz de realizar o rastreamento individual em situações com mais de um indivíduo por baia.

O resultado demonstrado por este trabalho confirma a possibilidade de utilização de imagens de simples câmeras monoculares para analisar o comportamento alimentar de suínos em granjas. Porém, para realizar essa análise de forma individual, é necessário o uso de algoritmos de rastreamento resilientes frente a intensas oclusões. Neste sentido, podem ser estudadas outras técnicas de visão computacional, como por exemplo o fluxo óptico, e filtros probabilísticos, como o de Kalman e o de Partículas.

Uma possível alternativa ao uso de algoritmos de rastreamento para realizar a análise de forma individual, é a utilização de marcos visuais como, por exemplo, tintas de diferentes cores que facilitem para o sistema a identificação de cada indivíduo, realizando a contagem individual de tempo de alimentação.

Note que o sistema que neste trabalho focou no comportamento alimentar, também pode ser futuramente utilizado para identificar outros tipos de comportamentos dos animais. Um exemplo é o nível de interação entre os animais, que pode sugerir a presença de animais aptos para a reprodução, ou a detecção de animais muito estáticos, que pode facilitar a detecção de animais doentes.

## REFERÊNCIAS

- ABDELALI, H.; ESSANNOUNI, F.; ABOUTAJDINE, D. Object tracking in video via particle filter. **International Journal of Intelligent Engineering Informatics**, v. 4, p. 340, 2016.
- AHRENDT, P.; GREGERSEN, T.; KARSTOFT, H. Development of a real-time computer vision system for tracking loose-housed pigs. **Computers and Electronics in Agriculture**, v. 76, n. 2, p. 169 – 174, 2011. ISSN 0168-1699.
- ALAMSYAH, D.; FACHRURROZI, M. Faster r-cnn with inception v2 for fingertip detection in homogenous background image. **Journal of Physics: Conference Series**, v. 1196, 2019.
- ANIL, S.; ANIL, L.; DEEN, J. Challenges of pain assessment in domestic animals. **Journal of the American Veterinary Medical Association**, v. 220, 2002.
- BOTTOU, L. Large-scale machine learning with stochastic gradient descent. In: **Proceedings of COMPSTAT'2010**. [S.l.]: Springer, 2010. p. 177–186.
- CHEN, P.-J. et al. Development of a management system with rfid and qr code for matching and breeding in taiwan pig farm. In: . [S.l.: s.n.], 2016. p. 1–5.
- COMANICIU, D. et al. Kernel-based object tracking. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 25, 2003.
- COMINOTTE, A. et al. Automated computer vision system to predict body weight and average daily gain in beef cattle during growing and finishing phases. **Livestock Science**, v. 232, 2020. ISSN 1871-1413.
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: **2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)**. [S.l.: s.n.], 2005. v. 1, p. 886–893 vol. 1.
- DASIOPOULOU, S. et al. Knowledge-assisted semantic video object detection. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 15, n. 10, p. 1210–1224, 2005.
- ESS, A. et al. Object detection and tracking for autonomous navigation in dynamic environments. **The International Journal of Robotics Research**, v. 29, n. 14, p. 1707–1725, 2010.
- EVERINGHAM, M. et al. The pascal visual object classes (voc) challenge. **International Journal of Computer Vision**, v. 88, p. 303–338, 2009.
- EVODAREK. The pros and cons of automation. **Designing Efficient Systems.**, 2012.
- FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861–874, 2006. ISSN 0167-8655. ROC Analysis in Pattern Recognition.
- GIRSHICK, R. et al. Region-based convolutional networks for accurate object detection and segmentation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 38, n. 1, p. 142–158, 2016.
- GUPTA, B. et al. Designing climate-smart future crops employing signal transduction components. **Elucidation of Abiotic Stress Signaling in Plants: Functional Genomics Perspectives vol 2.**, 2015.

- HANDCOCK, R. et al. Monitoring animal behaviour and environmental interactions using wireless sensor networks, gps collars and satellite remote sensing. **Sensors**, v. 9, 2009.
- HANSEN, M. F. et al. Towards on-farm pig face recognition using convolutional neural networks. **Computers in Industry**, v. 98, p. 145 – 152, 2018. ISSN 0166-3615.
- HARRIS, C.; STEPHENS, M. A combined corner and edge detector. In: **Proc. of Fourth Alvey Vision Conference**. [S.l.: s.n.], 1988. p. 147–151.
- HESSING, M. J. C.; TIELEN, M. J. M. The effect of climatic environment and relocating and mixing on health status and productivity of pigs. **Animal Science**, Cambridge University Press, v. 59, n. 1, p. 131–139, 1994.
- INDOLIA, S. et al. Conceptual understanding of convolutional neural network- a deep learning approach. **Procedia Computer Science**, v. 132, p. 679 – 688, 2018. ISSN 1877-0509. International Conference on Computational Intelligence and Data Science.
- JOSHI, K. A.; THAKORE, D. G. A survey on moving object detection and tracking in video surveillance system. **International Journal of Soft Computing and Engineering**, Citeseer, v. 2, n. 3, p. 44–48, 2012.
- KAMATE, S.; YILMAZER, N. Application of object detection and tracking techniques for unmanned aerial vehicles. **Procedia Computer Science**, Elsevier, v. 61, p. 436–441, 2015.
- KARPATHY, A. **CS231n: Convolutional Neural Networks for Visual Recognition**. 2019. Disponível em: <<https://cs231n.github.io/convolutional-networks/>>.
- KIM, J. et al. Depth-based detection of standing-pigs in moving noise environments. **Sensors**, v. 17, p. 2757, 2017.
- KIM, W.; CHO, Y. B.; LEE, S. Thermal sensor-based multiple object tracking for intelligent livestock breeding. **IEEE Access**, v. 5, p. 27453–27463, 2017.
- KOUTNÍK, J.; SCHMIDHUBER, J.; GOMEZ, F. J. Online evolution of deep convolutional network for vision-based reinforcement learning. In: **From Animals to Animats 13 - 13th International Conference on Simulation of Adaptive Behavior**. [S.l.]: Springer, 2014. (Lecture Notes in Computer Science, v. 8575), p. 260–269. ISBN 978-3-319-08863-1.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 60, n. 6, p. 84–90, 2017. ISSN 0001-0782.
- LAUBER, M.; VICTORIA. **Animal welfare for livestock producers / [author : Mariko Lauber]**. [S.l.]: Dept. of Primary industries Melbourne, Vic, 2007. ISBN 9781741998504.
- LERUSTE, H. et al. Effects of the observation method (direct v. from video) and of the presence of an observer on behavioural results in veal calves. **Animal**, v. 7, p. 1–7, 2013.
- LIN, T.-Y. et al. **Microsoft COCO: Common Objects in Context**. 2015.
- LIND, N. M. et al. Validation of a digital video tracking system for recording pig locomotor behaviour. **Journal of Neuroscience Methods**, v. 143, n. 2, p. 123 – 132, 2005. ISSN 0165-0270.

- MASELYNE, J. et al. Measuring the drinking behaviour of individual pigs housed in group using radio frequency identification (rfid). **Animal**, Cambridge University Press, v. 10, n. 9, p. 1557–1566, 2016.
- MASELYNE, J. et al. Validation of a high frequency radio frequency identification (hf rfid) system for registering feeding patterns of growing-finishing pigs. **Computers and Electronics in Agriculture**, v. 102, p. 10 – 18, 2014. ISSN 0168-1699.
- MASELYNE, J. et al. Validation of a high frequency radio frequency identification (hf rfid) system for registering feeding patterns of growing-finishing pigs. **Computers and Electronics in Agriculture**, Elsevier, v. 102, p. 10–18, 2014.
- MOTTRAM, T. Animal board invited review: precision livestock farming for dairy cows with a focus on oestrus detection. **Animal**, Cambridge University Press, v. 10, n. 10, p. 1575–1584, 2016.
- MOUNT, L. **The Climatic Physiology of the Pig**. [S.l.]: Edward Arnold, 1968. (Monographs of the Physiological Society). ISBN 9780713141405.
- NAIR, V.; HINTON, G. Rectified linear units improve restricted boltzmann machines vinod nair. In: . [S.l.: s.n.], 2010. v. 27, p. 807–814.
- NETO, T.; NOBRE, S. M. Bem-estar e produtividade de suínos. Universidade Federal do Pampa, 2015.
- NOLDUS, L.; SPINK, A.; TEGELENBOSCH, R. Ethovision: A versatile video tracking system for automation of behavioral experiments. **Behavior Research Methods, Instruments, Computers**, v. 33, p. 398–414, 2001.
- NOTT, S. B. Livestock farm management, electronics and automation. **American Agricultural Economics Association Meeting Pullman Washington**, 1979.
- PAPAGEORGIU, C.; POGGIO, T. A trainable system for object detection. **International Journal of Computer Vision**, v. 38, p. 15–33, 2000.
- PARK, H.; JEONG, H.; YOE, H. A study on the sow breeding management system for msy improvement based qr-code. **International Journal of Multimedia and Ubiquitous Engineering**, Citeseer, v. 9, n. 4, p. 179–186, 2014.
- PERNER, P. Motion tracking of animals for behavior analysis. In: ARCELLI, C.; CORDELLA, L. P.; BAJA, G. S. di (Ed.). **Visual Form 2001**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. p. 779–786. ISBN 978-3-540-45129-7.
- POMAR, C. A. et al. Applying precision feeding techniques in growing-finishing pig operations. **Revista Brasileira de Zootecnia**, scielo, v. 38, p. 226 – 237, 2009. ISSN 1516-3598.
- PORTO, S. M. et al. A computer vision-based system for the automatic detection of lying behaviour of dairy cows in free-stall barns. **Biosystems Engineering**, v. 115, n. 2, p. 184–194, 2013. ISSN 1537-5110.
- RAWAT, W.; WANG, Z. Deep convolutional neural networks for image classification: A comprehensive review. **Neural Computation**, v. 29, p. 1–98, 2017.

RAZAVIAN, A. S. et al. **CNN Features off-the-shelf: an Astounding Baseline for Recognition**. 2014.

SA, J. et al. Fast pig detection with a top-view camera under various illumination conditions. **Symmetry**, v. 11, p. 266, 2019.

SHAO, B.; XIN, H. A real-time computer vision assessment and control of thermal comfort for group-housed pigs. **Computers and Electronics in Agriculture**, v. 62, n. 1, p. 15 – 21, 2008. ISSN 0168-1699. Precision Livestock Farming (PLF).

SHI, J.; TOMASI. Good features to track. In: **1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 1994. p. 593–600.

SIMONYAN, K.; ZISSERMAN, A. **Very Deep Convolutional Networks for Large-Scale Image Recognition**. 2015.

SZEGEDY, C. et al. Going deeper with convolutions. In: **2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2015. p. 1–9.

SZEGEDY, C. et al. **Rethinking the Inception Architecture for Computer Vision**. 2015.

TILLET, R.; ONYANGO, C.; MARCHANT, J. Using model-based image processing to track animal movements. **Computers and Electronics in Agriculture**, v. 17, n. 2, p. 249 – 261, 1997. ISSN 0168-1699. Livestock Monitoring.

TORELLO, M. W. et al. An automated method for measurement of circling behavior in the mouse. **Pharmacology Biochemistry and Behavior**, v. 19, n. 1, p. 13 – 17, 1983. ISSN 0091-3057.

TURNER, L. et al. Monitoring cattle behavior and pasture use with gps and gis. **Can. J. Anim. Sci.**, 2009.

UIJLINGS, J. et al. Selective search for object recognition. **International Journal of Computer Vision**, v. 104, p. 154–171, 2013.

VEENMAN, C. J.; REINDERS, M. J. T.; BACKER, E. Resolving motion correspondence for densely moving points. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 23, n. 1, p. 54–72, 2001.

VESTEGEN BRASCAMP, V. d. H. Growing and fattening of pigs in relation to temperature of housing and feeding level. **Canadian Journal of Animal Science**, 1978.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: **Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001**. [S.l.: s.n.], 2001.

VOULODIMOS, A. S. et al. A complete farm management system based on animal identification using rfid technology. **Computers and Electronics in Agriculture**, v. 70, n. 2, p. 380 – 388, 2010. ISSN 0168-1699. Special issue on Information and Communication Technologies in Bio and Earth Sciences.

WELCH, G.; BISHOP, G. et al. **An introduction to the Kalman filter**. [S.l.]: Citeseer, 1995.

YILMAZ, A.; JAVED, O.; SHAH, M. Object tracking: A survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 38, n. 4, 2006. ISSN 0360-0300.

ZEILER, M. D.; FERGUS, R. **Visualizing and Understanding Convolutional Networks**. 2013.

ZOU, Z. et al. Object detection in 20 years: A survey. **ArXiv**, abs/1905.05055, 2019.