

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**WILLIAN OLIVEIRA PIRES**

**DETECÇÃO E CLASSIFICAÇÃO DE ALVÉOLOS EM FAVOS DE APIS  
MELLIFERA, UTILIZANDO ALGORITMOS DE DEEP LEARNING**

**MEDIANEIRA**

**2023**

**WILLIAN OLIVEIRA PIRES**

**DETECÇÃO E CLASSIFICAÇÃO DE ALVÉOLOS EM FAVOS DE APIS  
MELLIFERA, UTILIZANDO ALGORITMOS DE DEEP LEARNING**

**Detection and classification of alveoli in honeycomb of *Apis mellifera*, using  
deep learning algorithm**

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre em Tecnologias Computacionais para o Agronegócio do Programa de Pós Graduação em Tecnologias Computacionais para o Agronegócio da Universidade Tecnológica Federal do Paraná.

Orientador(a): Dr. Pedro Luiz de Paula Filho.

**MEDIANEIRA**

**2023**



[4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Esta licença permite remixe, adaptação e criação a partir do trabalho, para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es) e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



**Ministério da Educação  
Universidade Tecnológica do Paraná  
Campus Medianeira**



---

WILLIAN OLIVEIRA PIRES

**DETECÇÃO E CLASSIFICAÇÃO DE ALVÉOLOS EM FAVOS DE APIS MELLIFERA,  
UTILIZANDO ALGORITMOS DE DEEP LEARNING**

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestre Em Tecnologias Computacionais Para O Agronegocio da Universidade Tecnologica Federal do Paraná (UTFPR). Area de concentrasao: Tecnologias Computacionais Aplicadas A Produção Agrícola E Agroindústria.

Data de aprovação: 28 de Março de 2023

Dr. Arnaldo Candido Junior, Doutorado - Universidade Estadual Paulista - Unesp

Dra. Fabiana Costa De Araujo Schutz, Doutorado - Universidade Tecnologica Federal do Paraná

Dr. Regis Sivori Silva Dos Santos, Doutorado - Embrapa

Dr. Pedro Luiz de Paula Filho, Doutorado - Universidade Tecnologica Federal do Paraná

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 28/03/2023.

## RESUMO

A utilização de sistemas com machine learning em tarefas realizadas por humanos tem se tornado uma importante ferramenta, tornando atividades lentas e sujeitas a erro, mais rápidas e assertivas. Uma dessas atividades é a apicultura, um trabalho muito importante para preservação de abelhas e produção de alimento, mesmo com sua importância, este ainda é um processo muito manual com tarefas lentas e sujeitas a erro. Com o desenvolvimento de temas como machine learning e deep learning, surgiram novas ferramentas para o apicultor. O trabalho propõe desenvolver um modelo de deep learning que ajude no processo de contagem e classificação de alvéolos em favos, uma importante tarefa no manejo de colmeias. Para este desenvolvimento usou-se como base o projeto DeepBee e com ele foi separada a classe zangão e incluído no modelo, contemplando 8 classes (*capped*, *dontcare*, *drone*, *eggs*, *honey*, *larve*, *nectar* e *pollen*). Além de disponibilizar um algoritmo que permite facilitar o monitoramento de colmeias, através de arquivos. Foram utilizadas duas arquiteturas de modelos (MobileNet e ResNeXt) e a melhor apresentou uma f1-score de 95%.

**Palavras-chave:** Apicultura; machine learning; Rede Neural Convolutacional; Zangão.

## ABSTRACT

The use of systems with machine learning in tasks performed by humans has become an important tool, making slow and error-prone activities faster and more assertive. One of these activities is beekeeping, a very important job for the preservation of bees and food production, even with its importance, this is still a very manual process with slow and error-prone tasks. With the development of topics such as machine learning and deep learning, new tools have emerged for the beekeeper. The work proposes to develop a deep learning model that helps in the process of counting and classifying alveoli in combs, an important task in the management of hives. For this development, the DeepBee project was used as a base and with it the drone class was separated and included in the model, contemplating 8 classes (capped, dontcare, drone, eggs,honey,larve,nectar and pollen). In addition to providing an algorithm that facilitates the monitoring of hives, through files. Two model architectures were used (MobileNet and ResNeXt) and the best one presented an f1-score of 95%.

**Keywords:** Beekeeping; Machine learning; Convolutional Neural Network; Drone.

## LISTA DE FIGURAS

Figura 1 - Diferença do corpo do zangão.....	17
Figura 2 - Ovo, Larva e Pupa.....	18
Figura 3 - Exemplo de Colmeia Langstroth.....	21
Figura 4 - Revisão da Colmeia.....	22
Figura 5 - Representação de uma imagem digital.....	24
Figura 6 - Um sistema de visão computacional clássico e suas principais etapas.....	25
Figura 7 - Transformada de Hough.....	26
Figura 8 - Detecção do círculo.....	27
Figura 9 - Matriz de Confusão.....	30
Figura 10 - Exemplo do Perceptron.....	32
Figura 11 - Multilayer Perceptron.....	33
Figura 12 - Camada Convolutiva.....	34
Figura 13 - Rede Neural Convolutiva.....	35
Figura 14 - Arquitetura da MobileNet.....	35
Figura 15 - Arquitetura ResNeXt e os atalhos de bloco.....	36
Figura 16 - Funções de Ativação.....	37
Figura 17 - Metodologia.....	43
Figura 18 - Interface do DeepBee.....	44
Figura 19 - Exemplo de Imagens.....	44
Figura 20 - Exemplo dos Alvéolos.....	45
Figura 21 - Exemplo do Data Augmentations.....	46
Figura 22 - Exemplos de imagens de alvéolos de zangão.....	46
Figura 23 - Semelhança de alvéolos do zangão com outras classes.....	47
Figura 24 - Diferença entre o raio dos alvéolos.....	48
Figura 25 - Fluxograma do algoritmo.....	48
Figura 26 - Nova Arquitetura.....	50
Figura 27 - Formato do nome do arquivo.....	51
Figura 28 - Formato em Tabela.....	51
Figura 29 - Exemplo do processo de monitoramento.....	52
Figura 30 - Gráfico por linha.....	53
Figura 31 - Gráfico por quadrante.....	53
Figura 32 - Primeiro resultado do treino.....	54
Figura 33 - Resultado dos Modelos - com data augmentation.....	56
Figura 34 - Resultado da MobileNet com a classe zangão.....	57
Figura 35 - Resultado da ResNext com a classe zangão.....	58
Figura 36 - Métricas em relação ao raio.....	58
Figura 37 - Gráfico de contagem de classes.....	60
Figura 38 - Gráfico de posição das classes.....	60
Figura 39 - Gráfico de linhas do quadro.....	61
Figura 40 - Evolução de um quadro por 7 meses.....	62
Figura 41 - Gráficos seis meses depois.....	63
Figura 42 - Gráficos por região no quadro.....	63
Figura 43 - Gráficos por região no quadro - Figura (A) e Figura (B).....	64
Figura 44 - Imagens de quadros coletados no Brasil.....	65

## LISTA DE TABELAS

Tabela 1 - Resultado da Matriz - ResNeXt.....	55
Tabela 2 - Resultado da Matriz - MobileNet.....	55
Tabela 3 - Resultado da Matriz - MobileNet - Data Augmentations.....	56
Tabela 4 - Resultado da Matriz - ResNeXt - Data Augmentations.....	56
Tabela 5 - Matriz de confusão da MobileNet com zangão.....	59
Tabela 6 - Matriz de confusão da MobileNet.....	59

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>13</b>
<b>1.2</b>	<b>OBJETIVO GERAL.....</b>	<b>14</b>
1.2.1	Objetivos Específicos.....	15
<b>1.3</b>	<b>JUSTIFICATIVA.....</b>	<b>15</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO.....</b>	<b>16</b>
<b>2.1</b>	<b>Abelhas.....</b>	<b>17</b>
2.1.1	Ciclo de vida das abelhas.....	18
2.1.2	Ciclo: Rainha, Operárias e Zangão.....	19
<b>2.2</b>	<b>Produtos Comercializados das abelhas.....</b>	<b>19</b>
<b>2.3</b>	<b>Apiário.....</b>	<b>21</b>
<b>2.4</b>	<b>Colmeia.....</b>	<b>21</b>
<b>2.5</b>	<b>Processamento de Imagem.....</b>	<b>23</b>
2.5.1	Transformada de Hough.....	25
<b>2.6</b>	<b>Inteligência artificial e aprendizado de máquina.....</b>	<b>27</b>
2.6.1	Métricas.....	29
<b>2.7</b>	<b>Redes Neurais.....</b>	<b>31</b>
2.7.1	Redes Neurais Convolucionais (CNNs).....	33
2.7.2	Função de Ativação.....	37
2.7.3	Algoritmo Backpropagation.....	38
<b>2.8</b>	<b>Trabalhos Correlatos.....</b>	<b>40</b>
<b>3</b>	<b>MATERIAIS E MÉTODOS.....</b>	<b>42</b>
3.1	Ferramenta DeepBee.....	43
3.2	Treino de Modelos.....	44
3.2	Treino de Modelos.....	44
3.3	Classe Zangão.....	46
3.4	Monitoramento.....	50
<b>4</b>	<b>RESULTADOS.....</b>	<b>54</b>
4.1	Treinamento com a classe zangão.....	57
4.2	Monitoramento.....	59
<b>5</b>	<b>CONCLUSÃO.....</b>	<b>65</b>
	<b>REFERÊNCIAS.....</b>	<b>68</b>



## 1 INTRODUÇÃO

Um dos alimentos mais antigos do homem, o mel, passou por um processo de transformação da forma de sua aquisição e produção. Um processo que no início era necessário a procura de enxames de abelhas muitas vezes em locais de difícil acesso, passou por uma evolução até chegar a apicultura moderna, um processo mais seguro e eficaz (CAMARGO, 2002).

A apicultura surgiu no Egito antigo, há aproximadamente 4.400 anos, nesse período eram colocadas abelhas em recipientes cerâmicos, uma prática mais segura, mas ainda era considerada predatória, pois acabavam com as colônias ao retirar o mel e a cera (CRANE, 1999). O termo apicultura e colmeia surgiu na antiguidade clássica, para designar quem cria abelhas, do latim *apis* = abelha e *cultus* = cultivo (JENYNS, 2016).

A apicultura foi crescendo e adaptando as técnicas de manejo, como a criação de colmeias no período medieval, nessa época em Portugal foram criados os cortiços, estruturas feitas com casca da corticeira que abrigavam as abelhas, uma estrutura que permite aos apicultores recolher o mel excedente com o mínimo de danos às células das crias, por possuir um fundo aberto que permite o corte das pontas dos favos para colheita (CRANE, 1999).

No Brasil a apicultura começou em 1839, o padre Antonio Carneiro trouxe de Portugal 100 colônias de abelhas *Apis Mellifera*, mas apenas sete sobreviveram à viagem até o Rio de Janeiro. Imigrantes alemães e italianos inseriram outras subespécies de *Apis Mellifera* no Sul e Sudeste do Brasil, nesse período a criação de abelhas era vista como um hobby e para a produção de cera, existiam poucas técnicas de produção (ROCHA, 2018).

Em 1956 o manejo de colmeias já estava instalado no Brasil, mas devido a característica das abelhas e do clima do país, a produção de mel estava muito abaixo e ainda havia um número considerado de colmeias dizimadas por doenças. O professor Warwick Estevan Kerr, buscando reverter esse quadro, viajou para África em busca de rainhas que foram instaladas em Rio Claro, São Paulo (CAMARGO, 2002). A ideia era gerar estudos comparando abelhas africanas com as europeias,

avaliando a produtividade e a resistência, buscando espécies mais adequadas às condições brasileiras (ROCHA, 2018).

Devido a falhas de manejo, esses cruzamentos de abelhas acabaram gerando as abelhas africanizadas (CAMARGO, 2002), uma espécie mais agressiva e passou a ser considerado uma praga. Com o tempo essas abelhas passaram a ser utilizadas na apicultura, devido a sua produtividade e uma maior resistência a doenças que a europeia.

Atualmente a produção e comercialização de mel se tornou uma atividade bem lucrativa, segundo o IBGE (2020) foram produzidas 51.5 mil toneladas de mel no Brasil, sendo em valor de produção R\$ 621,5 milhões.

Mesmo com a importância ambiental e financeira do trabalho da apicultura, a América Latina carece de programas de incentivo e tecnologia no monitoramento de colmeias. A produção de mel na América Latina encontra-se em um estado de estagnação comparado ao mundo. Em 2016 a produtividade de mel na região apresentou um déficit de 460.000 toneladas e um déficit de 210.000 colmeias, em relação ao padrão mundial, muito devido à falta de programas e monitoramento para o apicultor (REQUIER F. ANTUNEZ, 2018).

Com o desenvolvimento de novas tecnologias e temas como reconhecimento de imagem, inteligência artificial, *machine learning* e *deep learning* começaram a surgir na Europa modelos e ferramentas que passaram a ajudar o apicultor em suas tarefas no manejo de colmeias.

Uma dessas ferramentas é o *DeepBee*, um software desenvolvido em Portugal, que permite contar a quantidade de alvéolos e identificar em sete classes: ovos, larva, tampado, outros, pólen, néctar e mel (ALVES., 2020). Nessa proposta busca-se aprimorar o software com comparativos em novas arquiteturas de modelos, inclusão de uma nova classe (zangão) e propor um módulo de monitoramento de colmeias. Como se trata de uma continuidade de trabalho e por falta de bases (imagens) nacionais, foi utilizada a base do *DeepBee*.

## 1.2 OBJETIVO GERAL

Desenvolver um modelo de detecção e classificação de alvéolos em favos de mel utilizando *deep learning* e aprimorando o *DeepBee* com inclusão do alvéolo de zangão no modelo, e um módulo de monitoramento.

### 1.2.1 Objetivos Específicos

- a) Testar novas arquiteturas de redes neurais convolucionais;
- b) Aprimorar o DeepBee para identificação dos alvéolos de zangão;
- c) Criar algoritmos para o monitoramento de colônias.

## 1.3 JUSTIFICATIVA

A apicultura tem uma importância econômica e ambiental, ela gera produtos como: cera, geléia real, apitoxina, própolis e o próprio mel. Além de atuar na preservação e criação de abelhas, que tem um importante papel na produção de alimentos, devido ao processo realizado por elas chamado polinização (CAMARGO, 2002).

Esses são alguns dos motivos da importância do manejo de colmeias, e uma etapa deste manejo é o controle de favos, importante para identificar a saúde da colmeia em que são feitas tarefas como: contagem de alvéolos no favo e identificação de larvas, pulpas, localização do mel no quadro e característica dos alvéolos, processo totalmente manual e lento. As tarefas visuais, como contagem de favos, têm sido otimizadas com modelos de reconhecimento de padrões, que extraem recursos de imagens com modelos com capacidade de generalização, permite criação de ferramentas importantes para tomadas de decisões e otimização de tempo.

Os algoritmos de visão computacional e aprendizado de máquina são cada vez mais utilizados no campo, impulsionando a 'agricultura inteligente', otimizando processos com novas ferramentas. Um exemplo dessas ferramentas é o *DeepBee* um software desenvolvido na Europa para auxiliar o apicultor no manejo de colmeias (ALVES, 2020). Portanto, esse trabalho visa desenvolver o tema de apicultura com ferramentas computacionais, o projeto será desenvolvido no Brasil em parceria com Portugal, como uma proposta de aprimoramento do software *DeepBee*.

## 2 REFERENCIAL TEÓRICO

O mel passou por muitas transformações na forma de se consumir e produzir, antigamente o alimento ingerido era uma mistura de mel, pólen, crias e cera. Não havia separação e nem conhecimento do produto, muitos morriam pelos enxames no processo de busca pelo mel (CAMARGO, 2002). O uso de colmeias silvestres surgiu a dez mil anos antes de Cristo, já o uso de potes para armazenamento do mel começou a 400 a.C pelos egípcios (FERNANDES, 2009).

Hoje em dia, a produção e a comercialização do mel tem se mostrado bem lucrativa. O maior produtor de mel mundial é a China, produz cerca de 500 mil toneladas de mel anualmente, um quarto da produção global (REUTERS, 2020). Estima-se que em 2019 os EUA tenham gerado U\$ 312.27 milhões com a produção de mel, já na união europeia 37% do mercado da produção de mel se divide em três países: Alemanha com U\$ 214 milhões, França com U\$ 184 milhões e Grécia com U\$ 133 milhões, seguidos de países como Itália, Espanha, Romênia, Polônia, Holanda, República Tcheca, Croácia e Portugal (TRADE, 2020).

Estima-se que os países da América do Sul representam 10% da produção mundial de mel, sendo Argentina a terceira maior produtora, ficando atrás de China e EUA (VIDAL, 2020). Segundo dados do IBGE de 2019 foram produzidas 45.980.621 toneladas de mel no Brasil, sendo em valor de produção R\$ 493.738 milhões (aproximadamente U\$ 94 milhões). O maior produtor foi o estado do Paraná com 7.228.717 toneladas produzidas (IBGE, 2019).

### 2.1 Abelhas

Estima-se que existam 20 mil espécies de abelhas no mundo, calcula-se que existam cerca de 300 espécies sem ferrão sendo mais sociáveis aos humanos, são da tribo Meliponiana (Hymenoptera) presentes na América do Sul, América Central, Ásia, Ilhas do Pacífico, Austrália, Nova Guiné e África. Abelhas-sem-ferrão são a grande maioria de espécies nativas encontradas no Brasil, já que essas abelhas são mais numerosas nas florestas tropicais (CAMPOS e PERUQUETTI, 2008).

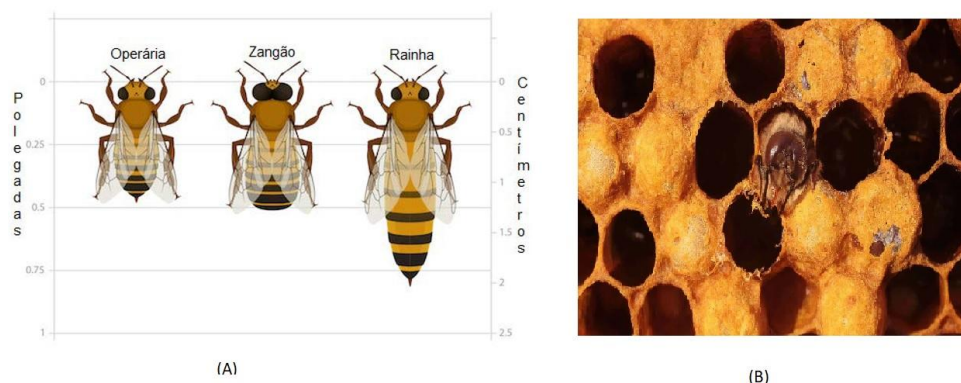
Entre as espécies com ferrão estão as do gênero *Apis*, chamadas como *Apis Melífera*, possuem origem europeia e são menos agressivas que as encontradas no

Brasil, medem de 12 mm a 13 mm de comprimento e apresentam pelos no tórax (BARBOSA, 2008). Já as abelhas mais agressivas do gênero *Apis*, são conhecidas como abelhas africanizadas, elas foram trazidas ao Brasil em 1839 devido a sua capacidade de produção de mel, essa espécie acabou evoluindo e se tornando o principal tipo de abelha de produção de mel no país, sendo denominada abelha africanizada brasileira (CPT, 2019).

Na formação de uma colmeia, as abelhas se dividem entre: rainha, operárias e zangões. A rainha possui a função de colocar ovos e manter a ordem na colmeia, já as operárias realizam a manutenção da colmeia realizando limpeza dos alvéolos e proteção da colmeia (WIESE e SALOME, 2020). O zangão tem como principal função a fecundação da rainha, e como atuam em mais de uma colmeia, acabam ampliando a diversidade genética na colmeia, a tornando mais resistente as doenças (CPT, 2019).

O alvéolo do zangão difere das outras abelhas, geralmente se encontra nos cantos dos quadros, e devido ao maior diâmetro do seu corpo, Figura 1 (A), possui alvéolos maiores, Figura 1 (B). É importante para o apicultor identificar as células da colmeia referente a zangão, porque os ácaros preferem se alimentar de zangão por demorarem mais para eclodir e seus alvéolos se tornam um bom local para ficar (WIESE e SALOME, 2020).

**Figura 1 - Diferença do corpo do zangão**



**Fonte: Adaptado BARBOSA et al. (2008)**

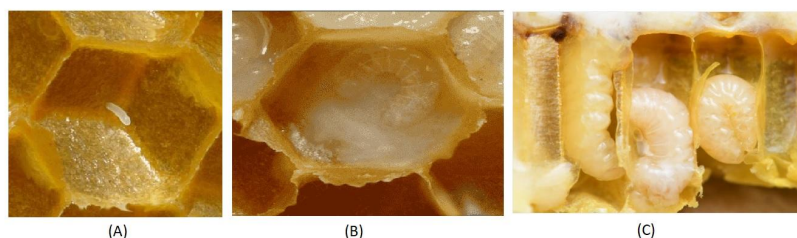
### 2.1.1 Ciclo de vida das abelhas

O ciclo de vida das abelhas está relacionado à sobrevivência na colmeia, as abelhas *Apis Melífera* são *eussociais*, são incapazes de viver na solidão e apenas uma parte específica da comunidade é capaz de reproduzir (SNODGRASS, 2016). Abelhas são *holometábolo*, que são espécies de insetos que se desenvolvem em quatro estágios: ovos, larvas, pupa e idade adulta (WIESE e SALOME, 2020).

Na fase de ovos (*Apis Mellífera* europeias) as rainhas podem pôr ovos que podem se tornar operárias, rainhas e zangões. São depositados em células na colmeia, e dependendo do indivíduo podem levar entre 14 e 24 dias para eclodir (SNODGRASS, 2016). Os ovos têm aproximadamente o tamanho de um grão de arroz e uma rainha coloca 2.000 a 3.000 ovos por dia, e ovos fertilizados se tornarão operárias ou rainha em potencial, já ovos não fertilizados se tornarão zangões, Figura 2 - A.

Após três dias os ovos se tornam larvas (Figura 2 - B), são cegas e não tem pernas, são alimentadas pelas operárias com geleias e permanecem na célula até atingirem o tamanho de uma pupa (CPT, 2019). Durante a fase da larva começam a gerar um casulo em torno de si e formar uma pupa (Figura 2 - C), nesse casulo a abelha se desenvolve em sua forma adulta, com surgimento de: asas, pernas, cabeça, tórax e abdômen (SNODGRASS, 2016). Em sua fase final a pupa se desenvolve em uma classe (operária, zangão e rainha) se tornando adulta e finalizando o ciclo. Novas rainhas levam cerca de 16 dias para emergir da célula do ovo, enquanto as abelhas operárias exigem entre 18 e 22 dias para se desenvolverem completamente e os zangões precisam de 24 dias (SNODGRASS, 2016).

**Figura 2 - Ovo, Larva e Pupa**



**Fonte: Adaptado Snodgrass (2016)**

### 2.1.2 Ciclo: Rainha, Operárias e Zangão

O ciclo de vida da abelha rainha pode durar até 4 anos, começando quando a colmeia está madura o suficiente para dar origem a outra colônia, em um local diferente. Nesse momento as abelhas operárias começam alimentar as larvas com a geleia real, estimulando o crescimento das larvas (SNODGRASS, 2016). Aproximadamente em 14 dias, a rainha emerge de sua célula rompendo a pupa, e após alguns dias sai para acasalar e formar uma nova colmeia (CPT, 2019).

As operárias constituem a maior parte de indivíduos na colmeia, em média existem 30.000 em cada colmeia. No inverno podem viver até 4 meses, mas na primavera, período de maior trabalho, não passam de um mês e meio de vida (SNODGRASS, 2016). São inférteis e realizam todas as tarefas da colmeia, em que suas atividades se dividem em relação a sua idade:

- 1-3 dias: acomodação dos ovos nas células;
- 3-6 dias: alimentação das larvas mais maduras;
- 6-10 dias: alimentação das larvas mais novas, são mais frágeis;
- 8-16 dias: recepção de pólen;
- 12-18 dias: confecção da cera para construção das células que formam o favo de mel;
- Dia 14 em diante: exploração, busca de néctar e pólen e proteção da colmeia.

Os zangões são abelhas machos, que eclodem de ovos não fertilizados, seu ciclo de vida tem início algumas semanas antes da colmeia começar a criar novas rainhas (WIESE e SALOME, 2020). O zangão ao passar pela fase da pupa sai da colmeia, e aguarda o acasalamento com a rainha, após isso ele morre, encerrando seu ciclo (CPT, 2019).

## 2.2 Produtos Comercializados das abelhas

O produto mais famoso produzido pelas abelhas é o mel, mas a colmeia produz outros produtos como: apitoxina, geleia real, pólen apícola, cera e própolis. Apitoxina é o veneno produzido pelas abelhas *A. Mellifera* utilizado para defesa da

colônia, o veneno é composto por: proteínas, enzimas, aminoácidos e lipídeos. No século XIX foram descobertas as propriedades biológicas da apitoxina, importante no tratamento de doenças, atuando como: anti-inflamatório, cicatrizante, neuroprotetor, antitumoral e analgésico. A comercialização desse produto não é indicada para pequenos produtores, devido à complexidade das técnicas utilizadas e na venda, já que só pode ser comercializado em farmácias e drogarias (SOUZA, 2020).

A geleia real é produzida pelas abelhas operárias mais novas (até 15 dias), sendo um produto muito importante para colmeia, já que serve de alimentação para as larvas em seu desenvolvimento e para rainha durante todo seu tempo de vida. No século XX as propriedades da geleia real começaram a ser avaliadas como matéria-prima em produtos de beleza e medicamentos como: anti-hipercolesterolêmica, antimicrobiana, anti-inflamatório, antitumoral e neuroprotetora (BEZERRA, 2018).

Já o pólen apícola é um produto retirado das flores e manipulado pelas abelhas, depositado nos alvéolos da colmeia e utilizado para alimentar larvas e abelhas adultas (a partir de 18 dias de idade). O produto é rico em nutrientes como: carboidratos, aminoácidos, lipídeos, vitaminas A, C, D, E, K e as do complexo B, minerais, fibras, além carotenoides, flavonoides, enzimas e fitoesteróis. Sendo um produto utilizado como alimento, e comercializado em diferentes formas: seco, misturado com mel, em cápsula ou tabletes (NASCIMENTO, PEDROSO e SOUZA, 2018).

A cera é produzida pelas abelhas operárias adultas (entre 12 a 18 dias) na construção dos abrigos e armazenamento de alimentos chamado de “favos”. A cera é também comercializada em indústria de produtos de beleza e medicamentos (NUNES, CORREIA-OLIVEIRA e SILVEIRA, 2018).

O própolis é produzido por abelhas na mistura da cera com resina das plantas, utilizada na colméia para evitar doenças e para fechar frestas no ninho, evitando entrada de correntes de ar frio. Atualmente, o própolis é utilizado para produtos de beleza e remédios como: anti-inflamatório, antimicrobiana, antioxidante, antiviral, cicatrizante e imunomodulatório (C.LUSTOSA, GALINDO e NUNES, 2008).

O mel é o produto mais comercializado da colmeia, sendo composto por açúcares, água, sais minerais e pequenas quantidades de vitaminas. Produzido pelas abelhas que colhem um líquido açucarado encontrado nas flores, chamado de



“néctar”. As abelhas cobrem os alvéolos com uma fina camada de cera para que o mel fique protegido. A cor, o sabor, o cheiro e a consistência do mel variam de acordo com as flores de onde foi retirado o néctar e com o clima. Rico em glicose e frutose, pode ser utilizado como alimento, adoçante e medicamentos (ESCOBAR e XAVIER, 2008).

### 2.3 Apiário

A criação de abelhas é feita em apiários, nome do local onde serão mantidas as colmeias, o apiário deve ser instalado em um local que existam plantas e flores que sejam fonte de néctar e pólen para as abelhas, esse local é conhecido como flora apícola. O apiário deve ser próximo às colmeias, facilitando as viagens das abelhas. O terreno deve ser plano e limpo evitando morros, por sofrerem muito com ventos, dificultando o trabalho das abelhas.

O apiário deve ficar entre 20 e 500 metros de uma fonte de água limpa, as entradas das colmeias devem ser voltadas para fonte de água, facilitando o início das atividades das abelhas (BARBOSA, 2008).

### 2.4 Colmeia

A colmeia possui diversos modelos, mas para seu uso no apiário, a colmeia mais indicada é o modelo de Langstroth, composta por tampa, sobrecaixa (meigueira), ninho, fundo e quadros, Figura 3.

**Figura 3 - Exemplo de Colmeia Langstroth**



Fonte: Adaptado BARBOSA et al. (2008)

As caixas devem ser compostas de madeira de boa qualidade, e podem ser feitas pelo apicultor. Na montagem das colmeias é necessário montar os quadros, feitos com arame e placa de cera alveolada, facilitando o trabalho das abelhas. Uma etapa importante da colmeia é seu povoamento, que pode ser feito comprando colmeias já povoadas, capturar enxames ou dividindo famílias fortes (BARBOSA, 2008).

Os cuidados de uma colmeia já populada começam pela espécie da abelha, africanizadas necessitam de dedicação e cuidados, não sendo indicado apiários com várias colmeias.

Entre os procedimentos que decidem se uma colmeia vai ter sucesso estão:

- Presença de alimento;
- Presença da rainha e sua postura;
- A força do enxame;
- Falta de espaço na colmeia;
- Doenças ou Pragas;
- Favos velhos;
- Peças danificadas.

Um item usado pelo apicultor na manutenção e cuidados da colmeia é o quadro, Figura 4, com ele são feitas as revisões na colmeia.

**Figura 4 - Revisão da Colmeia**



**Fonte: Adaptado SENAR (2009)**

Como essas revisões perturbam o trabalho das abelhas, devem seguir um protocolo (BARBOSA et al., 2008):

- Quinze dias depois da instalação do enxame, verificar os quadros;
- Verificar a melgueira a cada quinze dias, após a florada para verificar a produção de mel;
- Realizar as revisões das 08:00 às 11:00 horas da manhã, e das 15:00 às 17:30, horário que as operárias não estão na colmeia;
- Não fazer revisões em dia de chuva;
- Revisão deve ser feita por 2 pessoas, uma maneja o fumigador e a outra abre e revisa a colmeia;
- Realizar a revisão com calma, porém o mais rápido possível, evitando que a colmeia fique muito tempo aberta.

Avaliação dos quadros é importante para acompanhar o desempenho da colmeia, com ela é possível identificar presença de alimento (mel e polén) e crias (ovo, larva, pupa) é possível até identificar comportamentos na colmeia, como a falta de ovos uma semana após a troca da rainha, pode significar que a rainha não foi aceita na colmeia. Além de ajudar na identificação de doenças, pragas ou predadores.

## **2.5 Processamento de Imagem**

Uma imagem pode ser definida como uma função bidimensional  $f(x,y)$ , em que  $x$  e  $y$  são coordenadas espaciais e  $f$  é a amplitude ou brilho (nível de cinza) localizado em  $x$  e  $y$  (GONZALEZ e WOODS, 2018). Quando o  $x$ ,  $y$  e a intensidade do valor de  $f$  são todas quantidades finitas e discretas, a imagem é chamada de imagem digital. Uma imagem digital é uma imagem contínua em um arranjo matricial ( $L \times C$ ), em que cada elemento da matriz corresponde a um nível de cinza do pixel na imagem (GOMIDE, 2014), conforme a Figura 5.

**Figura 5 - Representação de uma imagem digital**

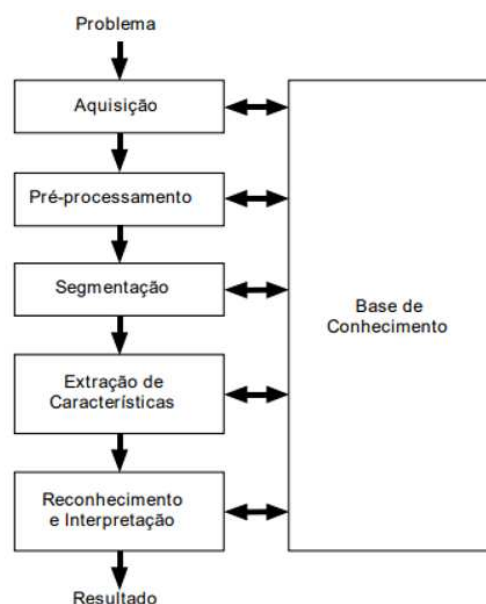
$$\left\{ \begin{array}{cccc} f(0,0) & f(0,1) & \dots & f(0,C-1) \\ f(1,0) & f(1,1) & \dots & f(1,C-1) \\ \vdots & \vdots & & \vdots \\ f(L-1,0) & f(L-1,1) & \dots & f(L-1,C-1) \end{array} \right\}$$

**Fonte: Adaptado Gomide(2014)**

Técnicas de processamento de imagem vem crescendo muito nos últimos anos, devido à variedade de aplicações que ela permite, como sensoriamento remoto, geoprocessamento, reconhecimento de imagem, restauração de imagem, entre outros. Uma área que também promoveu o crescimento do tema processamento de imagem foi a área de inteligência artificial, com disciplinas como *machine learning*, redes neurais e *deep learning*. Os elementos de um sistema de processamento de imagem podem ser divididos em aquisição, armazenamento, processamento e saída (GONZALEZ e WOODS, 2018).

Com o desenvolvimento de tecnologias como novas câmeras, arquitetura de computadores e a escalada de processamento paralelo (GPUs) tornaram-se viáveis tecnologias de visão computacional, buscando representar a visão humana. Um sistema de visão computacional deve ser capaz de adquirir, processar e interpretar imagens, conforme a Figura 6.

**Figura 6 - Um sistema de visão computacional clássico e suas principais etapas**



**Fonte: Adaptado Filho e Neto (1999)**

A primeira etapa é a aquisição de imagens, em que a fonte de aquisição varia com o domínio do problema, internet, fotografias, vídeo, sensores entre outros. Após a etapa de aquisição, é necessário avaliar a qualidade das imagens já que podem ocorrer problemas como: ruídos nos pixels, contrastes, brilho inadequado e informações faltando. Imagens com problemas irão gerar inconsistências no sistema, devido a isso é necessário aplicar técnicas de processamento de imagem visando uma maior qualidade da aplicação (GONZALEZ e WOODS, 2018).

A etapa de segmentação visa dividir uma imagem em unidades significativas na área de interesse da própria imagem, buscando simplificar ou mudar a representação da imagem para sua análise. Após a segmentação, busca-se extrair os descritores que permitirão identificar características da imagem que vão variar dependendo do domínio do problema, nessa etapa um conjunto de dados correspondem a imagem (ZANOTTA, FERREIRA e ZORTEA, 2019).

Na etapa de reconhecimento e interpretação ocorre o reconhecimento através das características e vão ser associadas a rótulos, e no processo de interpretação, determinar um significado aos objetos identificados. Na etapa de base de conhecimento, todas as etapas anteriores irão gerar informações que ajudarão na melhoria do sistema, e a ideia que essa base se retroalimente, o seu tamanho vai depender da característica do problema (GONZALEZ e WOODS, 2018).

### 2.5.1 Transformada de Hough

A transformada de Hough foi proposta em 1972 (ZANOTTA, FERREIRA e ZORTEA, 2019), uma técnica matemática que visa identificar formas geométricas em imagens, inicialmente tinha o objetivo de identificar retas, que com o passar do tempo foi possível identificar círculos e elipses.

Antes de explicar o algoritmo aplicado a círculos, é necessário explicar a identificação de retas. Uma reta pode ser representada matematicamente pela equação 1.

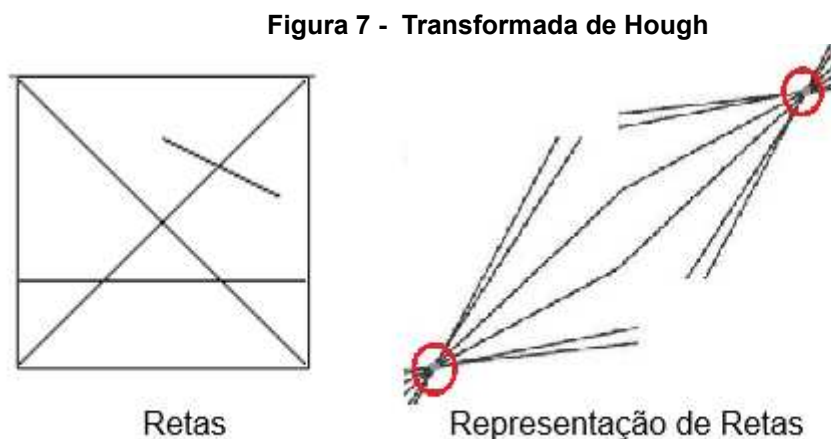
$$y = ax + b \quad (1)$$

Em que as características da reta são a inclinação ( $a$ ) e a intersecção ( $b$ ), em que uma reta pode ser representada como um ponto ( $a,b$ ), como os parâmetros são

ilimitados, tornando  $a, b$  infinitos quando a reta torna-se vertical, computacionalmente optou-se pela representação polar da reta, a equação 2.

$$p = x \cos \Theta + y \sin \Theta \quad (2)$$

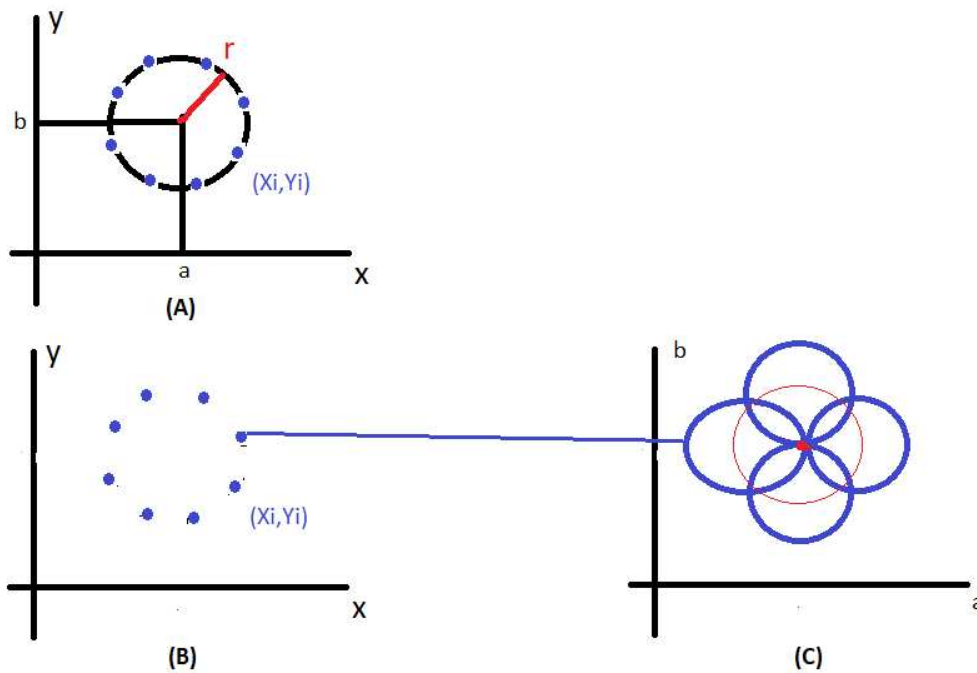
Em que um ponto  $(\Theta, p)$  e cada reta na imagem vai ser associada a esse ponto, infinitas retas passam por esse ponto (regiões destacadas com círculo vermelho na Figura 7, formando a senóide no plano Hough. Muitas senóides que intersectam num único ponto, representa a reta, (Figura 7).



**Fonte: Adaptado Gomide (2014)**

Já em relação a um círculo, pode ser representado matematicamente por  $x, y, r$ , na qual  $x, y$  referem-se ao ponto no centro e  $r$  ao raio, Figura 8 - (A). Para efeito computacional utilizando transformada de Hough 3-D, o algoritmo se divide em duas etapas: achar o centro e o raio do círculo. Para cada ponto  $(x, y)$  no círculo original ele pode definir um círculo de raio 1, conforme a Figura 8 - (B), esses círculos são armazenados em um vetor acumulador  $(A(a, b))$  em que só serão armazenados os ponto de intersecção de todos os círculos, que corresponde ao ponto central do círculo original (GOMIDE, 2014), (Figura 8 - (C)).

Figura 8 - Detecção do círculo



Fonte: Adaptado Smith (2018)

A transformada de Hough é uma das técnicas de processamento de imagem, que permite identificar figuras geométricas em imagens digitais. Outra área que também usa técnicas de processamento de imagem, é a inteligência artificial e aprendizado de máquina.

## 2.6 Inteligência artificial e aprendizado de máquina

Com o passar dos anos e a evolução da tecnologia, atividades feitas por humanos muitas vezes repetitivas e sujeitas a erro começaram a ser realizadas por máquinas, muitas delas utilizando inteligência artificial - IA. A IA pode ser definida como um estudo de como computadores podem realizar tarefas hoje realizadas por humanos (RICH, KNIGHT e NAIR, 1983), em busca de construir mecanismo que simule a capacidade humana com o menor erro possível. Essa área de estudo é recente no campo da ciência e engenharia, e segundo Warren McCulloch e Walter Pitts (1943) seu primeiro trabalho é da Segunda Guerra Mundial, entre os estudiosos que ajudaram no crescimento da área, estão Warren McCulloch e Walter Pitts (1943) com a proposta de um modelo de neurônio artificial, primeiro trabalho de IA, em

1956 Allen Newell e Hebert Simon criaram o Logic Theorist um programa de computador capaz de resolver problemas.

Um trabalho que ajudou na divulgação do tema IA, foi o artigo de 1950 chamado “Computing Machinery and Intelligence” escrito por Alan Turing, nele era proposto o teste de Turing. Atualmente o teste de Turing não é mais uma forma de avaliar inteligência (FALK, 2014), mas o teste consistia em um computador atuando em um interrogatório com um humano, o teste teria sucesso caso o interrogador não conseguisse diferenciar se a resposta vinha de uma máquina ou de um humano.

Para a máquina ter sucesso, deveria possuir as seguintes disciplinas:

- Processamento de Linguagem Natural;
- Raciocínio automatizado;
- Aprendizado de máquina;

Outras duas disciplinas que não estavam no teste, devido à falta de interação física desse teste, foram:

- Visão Computacional;
- Robótica.

Essas disciplinas representam a maior parte da área da IA (NORVIG e RUSSELL, 2004). Dentre as disciplinas citadas anteriormente, aprendizagem de máquina possui um destaque, devido à aprendizagem ser fundamental para a IA, em que definir aprendizagem de máquina é tão difícil quanto definir inteligência artificial, por possuir muitas definições, uma delas, diz que, aprendizagem de máquina é a capacidade de melhorar o desempenho de realização de tarefas por meio da experiência (FACELI, 2011).

Um item muito importante nesse processo é o princípio da indução, em que se obtém conclusões genéricas através de exemplos, e a criação de uma hipótese para resolver um problema. A hipótese deve ter sucesso com dados desconhecidos, possuir capacidade de generalização, quanto maior essa capacidade, melhor o



desempenho com dados desconhecidos (FACELI, 2011). As hipóteses induzidas são criadas no treinamento dos algoritmos, e esse algoritmo pode ser de diferentes tipos: clusterização, árvore de decisão, redes neurais, etc.

Os modelos de *machine learning* possuem tipos de aprendizados, que orientam o treinamento e o funcionamento dos algoritmos (FACELI, 2011). O primeiro aprendizado é denominado supervisionado, nele existe um supervisor que conhece a saída do modelo desejada, um exemplo são modelos de reconhecimento de imagem que são treinados com classes já conhecidas pelo modelo.

Já aprendizagem não supervisionada não existe o supervisor, não existem rótulos passados no treinamento, um exemplo são modelos de clusterização. Outra forma de aprendizado é por reforço nesse método existe a lógica da recompensa, uma ação considerada positiva será reforçada e a ação negativa será punida (HARRISON, 2019).

### 2.6.1 Métricas

No processo de desenvolvimento de um modelo de aprendizado de máquina, a etapa de avaliação é uma das mais importantes, com ela é possível determinar quando finalizar um treinamento e verificar o desempenho do modelo (HARRISON, 2019). E cada uma das métricas vão se basear nas técnicas utilizadas, existem métricas só para aprendizado supervisionado e outras para não supervisionado.

E para cada um desses métodos de aprendizagem existem ramificações, por exemplo, no aprendizado supervisionado existem modelos de regressão e classificação e cada um desses possuem suas métricas para avaliar seus modelos (NORVIG, 2021).

Os modelos de classificação visam prever a qual classe uma nova observação pertence, exemplo, prever a qual classe (mel, tampada, néctar, pólen, larva) o alvéolo observado pelo modelo pertencem. E sua avaliação ocorre comparando a classe prevista pelo modelo e a classe real, utilizando métricas.

Uma técnica muito utilizada em modelos de classificação, é matriz de confusão, essa matriz indica quantos exemplos existem em cada grupo: falso positivo (FP), verdadeiro positivo (VP), falso negativo (FN) e falso positivo (FP) (HARRISON, 2019), conforme figura 9 (a). Ela permite visualizar como está o

comportamento entre as classes, e com isso verificar qual classe possui maior taxa de acerto e qual classe possui maior erro, e com isso diminuindo a taxa de acerto do modelo (NORVIG, 2021).

Para analisar seu resultado será apresentado um exemplo, buscando classificar se o alvéolo tem mel ou não, foram passados 100 imagens, e seguindo a matriz foram passadas 80 imagens com mel e o modelo acertou 70 e errou 10, sugerindo como outra classe. Já a segunda linha da matriz, são passadas 20 imagens de alvéolos de outras classes e ele acertou 5 e errou 15, sugerindo como alvéolo com mel, conforme a figura 9 (b).

**Figura 9 - Matriz de Confusão**

	Classe 1	Classe 2
Classe 1	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Classe 2	Falso Positivo (FP)	Verdadeiro Negativo (VN)

(a)

	Mel	Outros
Mel	70	10
Outros	15	5

(b)

Fonte: Autoria Própria (2022)

É possível verificar com a matriz, que ela possui um alto VP, mas um baixo VN, o que significa que o modelo está com uma alta taxa de acerto para classificar alvéolos de mel, mas com uma baixa taxa para outras classes.

Caso a matriz de confusão não seja o suficiente para avaliar o desempenho do modelo, outra forma de mensurar é utilizando a acurácia, equação 3, que apresenta quantas amostras o modelo classificou corretamente, exemplo, 100 imagens de alvéolos de mel foram passadas para o modelo, ele previu 80 imagens corretamente, então sua acurácia é de 80% (NORVIG, 2021).

$$acuracia = \frac{VP + VN}{VP + VN + FP + FN} \quad (3)$$

Para verificar os casos positivos identificados, quantos são realmente da classe positiva é utilizada a métrica revocação ou também conhecida como *recall*, essa métrica é definida pela razão entre a quantidade de exemplos classificados

positivos e a quantidade real da classe positiva (CARVALHO, 2021), conforme a equação 4.

$$recall = VP / VP + FN \quad (4)$$

Outra métrica é precisão que é calculada pela razão entre a quantidade de exemplos classificados como positivo e o total de exemplos classificados como positivo, equação 5, por exemplo, se um modelo tem precisão de 80% para classificar alvéolos de mel, significa que 100 imagens classificadas na classe mel, somente 80 serão classificadas corretamente.

$$precisão = VP / VP + FP \quad (5)$$

Uma métrica também muito utilizada em modelos de classificação é f1-score, por utilizar duas outras métricas em seu cálculo, a recall e a precisão (HARRISON, 2019). A f1-score é definida pela média harmônica entre recall e a precisão, conforme equação 6. Quanto maior o f1-score maior será precisão e a recall, portanto é uma métrica que resume bem o desempenho do model (CARVALHO, 2021).

$$f1 - score = 2 * (precisão * recall / (precisão + recall)) \quad (6)$$

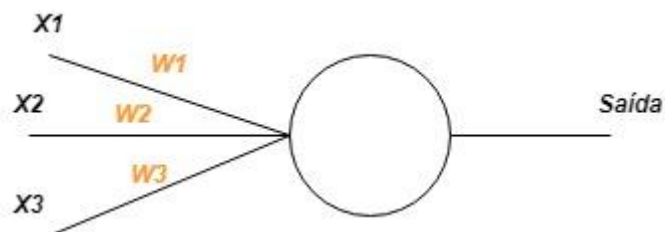
## 2.7 Redes Neurais

Um dos modelos mais utilizados do aprendizado de máquina são as redes neurais, algoritmos projetados para representar a maneira como o cérebro realiza uma tarefa em particular, sendo criada utilizando componentes eletrônicos ou simulada por programação (HAYKIN, 2001). A origem das redes neurais artificiais vem da biologia com o estudo do cérebro humano, buscando representar o neurônio biológico.

Existem diferentes tipos de arquitetura de redes, a estrutura mais simples é o Perceptron, esse modelo foi desenvolvido entre 1950 e 1960 pelo cientista Frank Rosenblatt (HAYKIN,2001). Um Perceptron é um modelo matemático que recebe

várias entradas  $x_1$ ,  $x_2$ , . . . e possui uma única saída, buscava representar um neurônio biológico, conforme a Figura 10.

Figura 10 - Exemplo do Perceptron



Fonte: Adaptado HAYKIN (2001)

A Figura 10 apresenta três entradas  $x_1$ ,  $x_2$  e  $x_3$ , Rosebenblatt propôs os pesos  $w_1$ ,  $w_2$  e  $w_3$  para expressar a importância de cada entrada e calcular o valor de saída (FACELI, 2011). A regra para calcular o valor de saída ocorre através de uma soma ponderada do produto das entradas  $x_j$  e seus pesos  $w_j$ , o valor final passa por um limiar para determinar o valor de saída, conforme a equação 7.

$$0 \text{ se } \sum_j w_j x_j \leq \text{limiar}$$

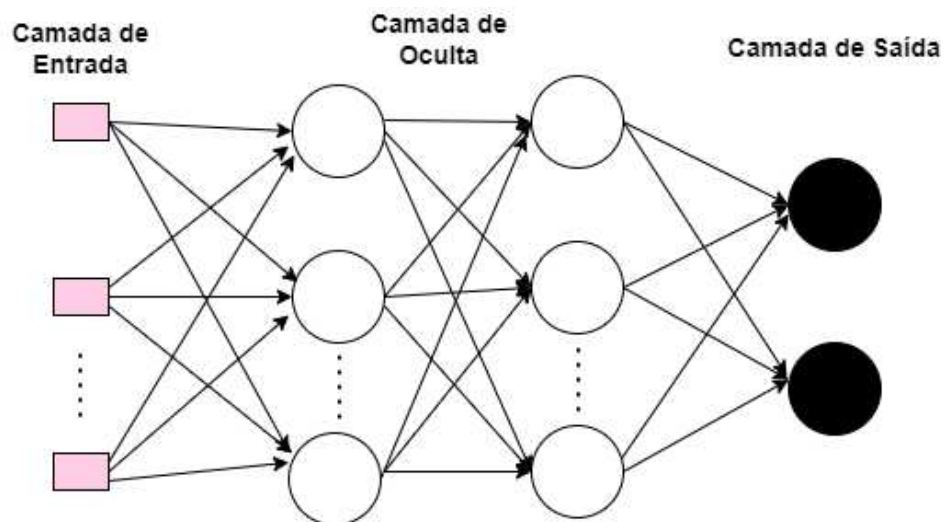
$$1 \text{ se } \sum_j w_j x_j > \text{limiar} \quad (7)$$

O limiar determina o valor de saída, no caso se o somatório do produto entre valores de entrada e seus pesos for maior que o limiar a saída vale 1, caso contrário vale 0. É um modelo matemático mais simples, dentre as redes neurais, mas já permite realizar tarefas menos complexas (SILVA; SPATTI; FLAUZINO, 2015).

Apenas um neurônio não é o suficiente para realizar tarefas complexas, um perceptron é incapaz de realizar classificação não linear, isso impede do modelo realizar tarefas mais complexas, como representar o comportamento da porta XOR, que verifica se os valores em suas entradas são diferentes (HAYKIN, 2001). Por isso existem diferentes arquiteturas de redes neurais que fazem uso de mais de um

neurônio para realizar previsões mais elaboradas (CARVALHO, 2021). Um exemplo de estrutura de redes é o multilayer perceptron (MLP) que é uma rede neural composta de mais de um neurônio (perceptron). São compostos por uma camada de entrada, uma camada de saída e a uma ou mutiplas camadas ocultas, conforme a Figura 11.

**Figura 11 - Multilayer Perceptron**



**Fonte: Adaptado HAYKIN (2001)**

### 2.7.1 Redes Neurais Convolucionais (CNNs)

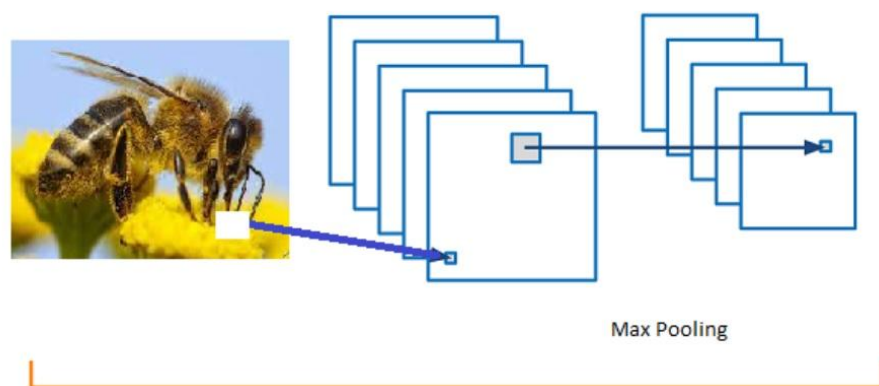
O MLP foi um modelo muito utilizado para pesquisa de aprendizado de máquina e em problemas de classificação, atualmente a arquitetura de rede que vem apresentando melhores resultados em problemas de classificação de imagens, são as redes convolucionais (CARVALHO, 2021). As redes neurais convolucionais (CNNs) são redes profundas que podem ser usadas para classificar imagens, reconhecimento de objetos em cenas, identificar rostos e também podem ser utilizadas em processamento de linguagem natural.

A primeira CNN foi proposta por Yann LeCun em 1989 chamada de LeNet, ela possui uma camada de entrada, muitas camadas ocultas e uma camada de saída (CARVALHO, 2021). As CNNs são consideradas modelos estado da arte em reconhecimento de imagem, para identificar um objeto na imagem, a CNN recebe a

imagem que é um objeto tridimensional, já que uma imagem é matriz de pixel que possui altura, largura e a codificação de cor, no caso da Figura 12 é uma RGB (red, green e blue) (HAYKIN, 2001). A primeira etapa de uma CNN é a camada convolucional geralmente junto com a camada de max pooling, é camada da rede composta por mapas de características, a camada de max pooling é usada para reduzir as dimensões dos dados (NUMPYDL, 2015), conforme a Figura 12.

O mapa de características se desloca em torno de uma imagem, realizando a operação de multiplicação de matrizes, no caso o filtro e a imagem, resultando no chamado mapa de ativação, que tem como objetivo identificar características da imagem como: bordas, retas, cores simples e curvas nas primeiras camadas, já nas últimas camadas da rede, consegue identificar objetos mais complexos (SILVA, SPATTI e FLAUZINO, 2015).

**Figura 12 - Camada Convolucional**

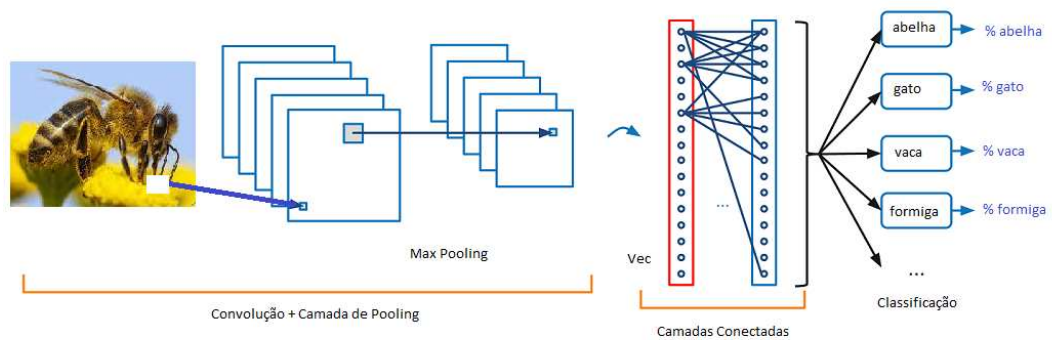


**Fonte: Adaptado Silva, Spatti e Flauzino (2015)**

A imagem passa pelas camadas da rede, alterando as dimensões da matriz resultante da imagem até gerar uma série de probabilidades na camada saída, em que cada probabilidade representa a possibilidade de uma classe (abelha, gato, vaca, ...) a com maior probabilidade será a classe definida, conforme a Figura 13.

Nas arquiteturas de redes neurais convolucionais existem diferentes redes com alterações em suas estruturas em busca de uma maior taxa de acerto e desempenho, entre elas pode se citar a VGG, ResNet, AlexNet, GoogleNet, MobileNet, ResNeXt, entre outras (CARVALHO, 2021).

**Figura 13 - Rede Neural Convolucional**



**Fonte: Adaptado Silva, Spatti e Flauzino (2015)**

A MobileNet é uma rede neural convolucional que possui uma arquitetura simplificada, utiliza convoluções separáveis para construir uma rede mais leve projetada para funcionar em aplicativos móveis, possui um código fonte aberto proposto pelo Google (HOWARD et al., 2015). A Arquitetura da rede (Figura 14) é projetada para otimizar taxa de acerto permitindo usar os recursos restritos de dispositivos móveis, podendo ser utilizada em problemas de classificação, detecção, embeddings e segmentação.

**Figura 14 - Arquitetura da MobileNet**

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool $7 \times 7$	$7 \times 7 \times 1024$
FC / s1	$1024 \times 1000$	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

**Fonte: Xie et al. (2018)**

A ResNeXt é uma rede neural convolucional utilizada principalmente em problemas

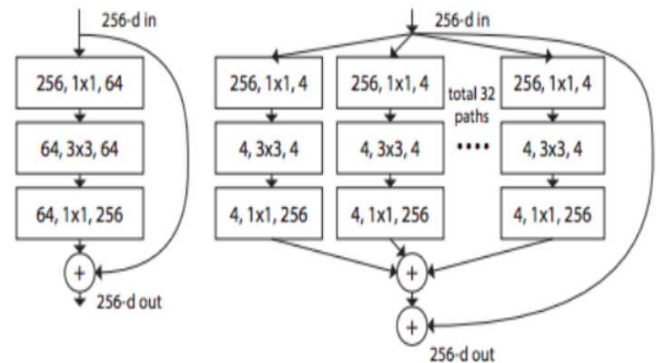
de classificação. Para melhorar a acurácia desse modelo, uma solução é aumentar o número de camadas, mas aumentando a complexidade da rede mais poder computacional se faz necessário, a idéia da ResNeXt é aumentar acurácia sem aumentar a complexidade (XIE et al., 2018). Arquitetura da ResNeXt, Figura 15 (a).

A ResNeXt utiliza técnicas herdadas de modelos como Resnet, VGG e Inception, em que inclui atalhos no bloco anterior para o próximo, empilhando camadas, essa estrutura criar uma conexão residual, lidando com um problema que o algoritmo de aprendizado da rede esta sujeito que é dissipação do gradiente, essa estratégia melhora o treinamento do modelo, Figura 15 (b).

**Figura 15 - Arquitetura ResNeXt e os atalhos de bloco**

stage	output	ResNet-50	ResNeXt-50 (32×4d)
conv1	112×112	7×7, 64, stride 2	7×7, 64, stride 2
		3×3 max pool, stride 2	3×3 max pool, stride 2
conv2	56×56	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128, C=32 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3	28×28	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256, C=32 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4	14×14	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512, C=32 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
conv5	7×7	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 1024 \\ 3 \times 3, 1024, C=32 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params.		$25.5 \times 10^6$	$25.0 \times 10^6$
FLOPs		$4.1 \times 10^9$	$4.2 \times 10^9$

(A)



(B)

Fonte: Xie et al. (2018)

A ResNeXt possui a mesma estrutura da ResNet, mas com uma camada de empilhamento em paralelo e possui um módulo de divisão e transformação semelhante à rede Inception mas compartilha hiperparâmetros entre as camadas. E com essas camadas em paralelo e elementos de outras redes foi constatado que o aumento da cardinalidade do modelo permite uma melhora dos resultados sem aumentar a complexidade da rede (XIE et al., 2018).



### 2.7.2 Função de Ativação

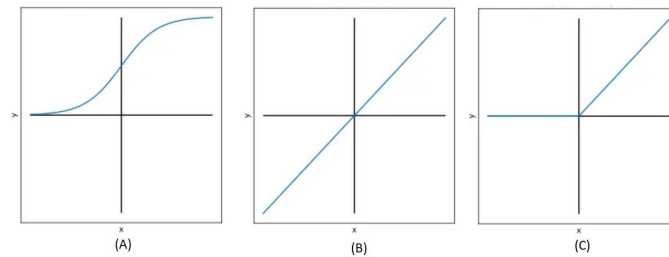
Modelos de redes neurais realizam transformações lineares nos dados, ao se deslocar entre os neurônios, para permitir as redes realizarem processamento não linear foram incluídas as funções de ativação. Com a inclusão da função de ativação, as redes deixam de ser apenas modelos lineares, para realizar tarefas mais complexas, como classificação de imagens e tradução de idiomas (KUTUK, 2022).

As funções de ativação habilitam o processamento não-linear das redes neurais, além disso são elas que determinam se um neurônio deve ser ativado ou não, tarefa importante para o aprendizado da rede. Existem diferentes tipos de funções de ativação, em que se tem finalidades específicas, dependendo do problema que será utilizado a rede neural (NORVIG, 2021).

Uma função de ativação muito utilizada é a Sigmoide, conforme a Figura 16 (a), devido ao seu intervalo de saída ser entre  $[0,1]$ , indicada para problemas de classificação, quando deve-se prever uma classe e sua probabilidade entre 0 e 1, facilita a análise (NORVIG, 2021).

Outra função de ativação é a linear, conforme a Figura 16 (b), em que sua ativação é proporcional à valor entrada, é utilizada para problemas mais simples, quando usada sem outras funções de ativação. A função de ativação geralmente mais utilizada para problemas de classificação é a ReLU, Figura 16 (c), por ser uma função computacionalmente mais leve, ela retorna 0 para números negativos, aumentando a velocidade da aprendizagem da rede e são geralmente utilizadas nas camadas escondidas (Kütük, 2022).

**Figura 16 - Funções de Ativação**



**Fonte: Autoria Própria (2022)**

### 2.7.3 Algoritmo *Backpropagation*

No aprendizado supervisionado existe o algoritmo *Backpropagation*, utilizado em redes de multicamadas, um trabalho que se propôs a explicar sobre o algoritmo *Backpropagation*, foi proposto em 1986 por Rumelhart, Hilton e Williams (SILVA, SPATTI e FLAUZINO, 2015) em *Learning Internal Representations by Error Propagation*. O *Backpropagation* possui duas fases, para frente (*forward*) e para trás (*backward*), na etapa para frente o objetivo é propagar os valores de entrada pela rede através de multiplicação de vetores até chegar na camada de saída. Um dos objetivos do algoritmo é minimizar o erro quadrático da rede neural. A soma do erro quadrático de cada neurônio da última camada é dada pela equação 8.

$$\epsilon^2 = \sum_{i=1}^{ns} \epsilon_i^2 \quad (8)$$

em que:

$$\epsilon = d_i - y_i$$

$d_i$  : saída desejada do i-ésimo elemento da última camada da rede;

$y_i$  : saída do i-ésimo elemento da última camada da rede;

$ns$  : número de neurônios da última camada da rede.

O algoritmo *backpropagation* realiza uma correção em  $\Delta w_{ij}(t)$  a peso sináptico  $w_{ij}(t)$ , por:

$$\Delta w_{ij}(t) = \eta * \delta(t) \cdot X_i(t) \quad (9)$$

em que:

$\Delta w_{ij}(t)$  : correção aplicada ao peso sináptico  $w_{ij}(t)$ ;

$\eta$  : taxa de aprendizagem;  
 $\delta_j(t)$  : erro produzido pelo neurônio  $t$ ;  
 $X_i(t)$  : entrada  $i$  do neurônio  $t$

Se o neurônio for da camada de saída, o erro é definido por (HAYKIN, 2001):

$$\delta(t) = \epsilon_j(j) \cdot \varphi^l |v_j(t)| \quad (10)$$

em que:

$\epsilon_j(j)$  : erro definido pela equação 8;

$$v_j(t) = \sum_n^i = X_i \cdot W_{ij} \text{ em que calculado com a equação 9;}$$

$h$  : representa o índice de iteração;

$\varphi^l |v_j(t)|$  : é a derivada aplicada  $v$ .

Caso contrário, o erro é definido por:

$$\delta(t) = \varphi^l |v_j(t)| \cdot \sum_k^k \delta_k \cdot w_{kj} \quad (11)$$

na qual:

$k$  : é o neurônio da camada que segue a camada de  $j$

$\delta_k$  : erro produzido no neurônio  $k$

Os pesos são atualizados, seguindo a equação:

$$w_{ij}(t + 1) = w_{ij}(t) + \Delta w_{ij} \quad (12)$$

Com as equações 9 e 12, tem-se:

$$w_{ij}(t + 1) = w_{ij}(t) + \eta \cdot \delta(t) \cdot X_i(t) \quad (13)$$

Os pesos da rede são iniciados arbitrariamente no intervalo (0,1), sendo o treinamento executado via o algoritmo backpropagation podendo ser em duas formas: a primeira ocorre o ajuste de pesos em todas as camadas, ocorrendo a convergência para cada padrão, sendo repetido até a total convergência da rede, em que o erro quadrático seja menor que a tolerância preestabelecida para os padrões. A segunda forma é o mesmo processo da primeira, mas se faz apenas uma iteração padrão por vez (RICH, KNIGHT e NAIR, 1983).

## 2.8 Trabalhos Correlatos

Nessa seção serão apresentados trabalhos relacionados ao tema machine learning aplicadas a apicultura, com ferramentas que ajudam no manejo de colmeias. Um exemplo é projeto proposto por Yoo (2021) em que utilizou modelos de aprendizagem de máquina para identificar espécies de abelhas, buscando entender a dinâmica populacional e a saúde da colmeia. Foi utilizado um conjunto de dados de imagem de onze espécies diferentes de Bumblebee (*Bombus*), mel de abelha (*Apis Mellifera*) e Carpenter bee (*Xylocopa* sp.) do iNaturalist, sendo treinados quinze modelos, o com melhor resultado foi a ResNet101 com precisão de 93,66%.

Outro trabalho que utilizou técnicas de reconhecimento de imagem no manejo de colmeias foi CombCount um script em python que detecta automaticamente favos sem cobertura para acelerar a medição de ninhada e mel na colmeia, o programa é gratuito, e se mostrou 3,2 vezes mais rápido que o Imagej, ferramenta com mesma função, o CombCount utilizou biblioteca de processamento de imagem Opencv (COLIN et al., 2018).

O surgimento de áreas como *machine learning* e deep learning alteraram métodos e técnicas utilizadas na computação, uma das áreas que mais evoluiu, foi o campo de reconhecimento de imagem, e muitos trabalhos passaram a utilizar essas técnicas no manejo de colmeias. Um exemplo de trabalho é o sistema de monitoramento de colmeias de *Apis mellifera*, proposto por Katarzyna Bozek (BOZEK et al., 2021), utilizando vídeo de alta resolução em um fundo de favo de mel. Para isso foram utilizados modelos de deep learning como redes neurais convolucionais para identificar automaticamente as posições dos alvéolos, e das ninhadas e a orientação, e dimensões do corpo da abelha.

O modelo apresentou acurácia 90% na identificação da largura corpo e 90% na orientação, além disso, o modelo conseguiu identificar comportamentos individuais das abelhas que podem ser objeto de estudo (BOZEK et al., 2021).

Além de utilizar imagens e vídeo para analisar abelhas, existem estudos (NGO et al., 2021) que desenvolveram um sistema para identificar o transporte do pólen em uma colmeia utilizando vídeo em alta resolução, foi desenvolvido um modelo de rede neural convolucional para identificar abelhas *Apis Mellífera* portadoras de pólen e não portadoras.

O modelo apresentou f1-score de 94%, e conclui que alterações do clima como temperatura e luminosidade tiveram impacto na colmeia. Utilizando vídeo para o manejo de colmeias existe também o projeto BJERGE et al. (2019) que é um sistema capaz de monitorar infestação de ácaro *Varroa* que destroem colmeias, utilizando gravações de abelhas *Apis Mellífera* por 5 a 20 minutos, foi desenvolvido uma estrutura para coleta de vídeo e utilizado um modelo de deep learning, foram analisados 1.775 abelhas e 98 ácaros, o modelo apresentou uma f1-score de 97% para contagem de abelhas e 91% para contagem de ácaros (BJERGE et al., 2019).

Uma importante etapa no monitoramento de colmeias é a identificação de parasitas, o trabalho BERKAYA, SORA e GUNAL (2017) se propõe a identificar além da *Varroa*, outros parasitas como formigas e besouro diferenciando das abelhas, utilizando modelos de deep learning aplicados em imagens, ao todo 19.393 imagens, o modelo apresentou precisão de 99% na identificação de anomalias em colmeias (BERKAYA, SORA e GUNAL, 2017).

Outro projeto que faz utilização de *deep learning* é o software open source DeepBee, uma ferramenta que visa auxiliar o apicultor na tarefa de controle de colmeia, desenvolvido no Instituto Politécnico de Bragança em parceria com a Universidade Tecnológica Federal do Paraná, com ele é possível identificar os alvéolos e fazer sua contagem e classificar em sete classes: ovos, larva, tampado, polén, néctar, mel e outros.

Para desenvolver o software foram testadas treze arquiteturas de rede neural, e como critério de escolha foram utilizados além da acurácia do modelo, outros critérios como: tempo de treino, tamanho do modelo, tempo de carregamento em memória do modelo, número de épocas de treino e tempo necessário para classificar 100 imagens.

O modelo escolhido foi a MobileNet que apresentou acurácia de 94,3%, para criar a base de imagens o modelo utiliza técnicas de segmentação e identificação de círculos em imagem (DJEKOUNE, MESSAOUDI e AMARA, 2017), fazendo uso de bibliotecas, um exemplo é Opencv (ALVES et al., 2020).

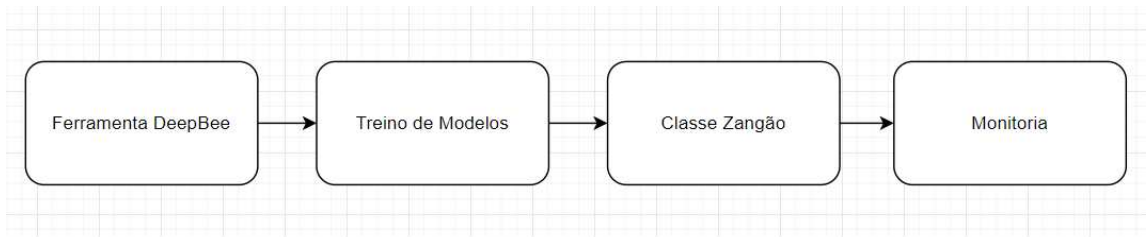
O DeepBee disponibiliza uma interface para o apicultor para verificar as sugestões do modelo com às sete classes (separadas por cor), e o número de células identificadas, caso a sugestão esteja incorreta o software permite realizar a correção e retreinar o modelo.

Um trabalho mais recente é o reconhecimento de imagem, proposto por NART (2022), usando redes neurais convolucionais para classificação de subespécies de abelhas *Apis mellifera*, em que foram utilizadas 9.887 imagens de asas de 7 espécies de abelhas e um híbrido para o treinamento. Foram avaliados os modelos ResNet 50, MobileNet V2, Inception Net V3 e Inception ResNet V2. Os modelos apresentaram acurácia acima de 92% (NART, 2022).

### **3 MATERIAIS E MÉTODOS**

Nessa seção será apresentada a metodologia do desenvolvimento do modelo de detecção e classificação de favos, com a classe zangão, e o algoritmo de monitoramento de colônias. Esse trabalho dará continuidade ao projeto do DeepBee, sendo uma das etapas principais o entendimento do software, e buscar aprimorar a ferramenta. Como primeira melhoria, foi proposto a realização de um comparativo com um modelo não existente, no caso a ResNeXt de quando DeepBee foi criado, além de recorrer a bibliotecas diferentes para o treinamento.

Em sequência foi feita a inclusão de uma nova classe (zangão) ao modelo, apresentando todo processo de treino e aquisição da base, e a última etapa foi desenvolver um algoritmo que junto ao DeepBee busca ajudar o apicultor na monitoria da colmeia, utilizando regiões do quadro. Portanto a metodologia acompanha quatro etapas, conforme a Figura 17.

**Figura 17 - Metodologia**

**Fonte: Autoria Própria (2022)**

### 3.1 Ferramenta DeepBee

A primeira etapa foi estudar e utilizar o software DeepBee, já que um dos objetivos do trabalho é o aprimoramento dessa ferramenta, já citado em trabalhos correlatos, o DeepBee é um software de código aberto, que disponibiliza um modelo de classificação e detecção de favos e possui uma interface para manipular as previsões do modelo.

É possível corrigir previsões incorretas e retreinar o modelo, através da interface, Figura 18. No DeepBee foram treinados treze tipos de modelos de aprendizado de máquina, e o que apresentou maior assertividade e desempenho computacional foi a MobileNet, com F1-score de 94,3%.

A MobileNet foi treinada com imagens que foram coletadas em apiários em Portugal, nas cidades de Bragança e na região de Algarve. Após entendimento da ferramenta e seu uso, a próxima etapa foi comparar o modelo do DeepBee com um modelo mais recente.

**Figura 18 - Interface do DeepBee**



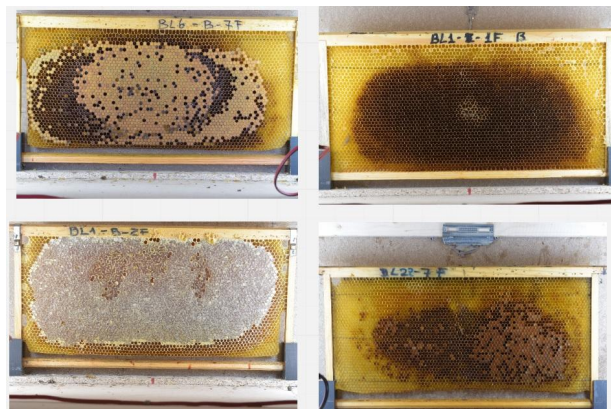
**Fonte: ALVES et al. (2020)**

### 3.2 Treino de Modelos

Para avaliar o DeepBee foi realizado um novo treinamento com o modelo da ResNeXt, uma arquitetura mais recente que a MobileNet. Para o desenvolvimento ResNeXt foi criado um código com o PyTorch, para comparar com o Modelo do DeepBee, foi também desenvolvido um modelo com a arquitetura MobileNet em linguagem python e com o Tensorflow.

A base utilizada foi a disponibilizada pelo DeepBee, devido a dificuldade de coletar imagens no período da pandemia. A base possui 2.206 imagens de quadros de favos e foram baixadas pelo link do github (ALVES et al., 2020), exemplo das imagens Figura 19.

**Figura 19 - Exemplo de Imagens**



**Fonte: Autoria Própria (2022)**



Como visto as imagens são de quadros de colmeias, mas para o modelo realizar a previsão é preciso separar os alvéolos do quadro, para esse processo foi utilizado o software DeepBee que já detecta e separa os alvéolos do quadro, Figura 20.

**Figura 20 - Exemplo dos Alvéolos**



**Fonte: Autoria Própria (2022)**

Com as imagens, a próxima etapa foi realizar o treino e avaliar o desempenho dos modelos, foram realizados dois treinos, com cada um dos modelos: ResNeXt e MobileNet.

O primeiro foi apenas utilizando as imagens geradas pelo DeepBee, o segundo foi utilizada a técnica de data augmentation. Os modelos que serão treinados são redes neurais convolucionais, e nesses modelos o quesito quantidade de amostras utilizadas no treino, é um dos critérios para um bom desempenho do modelo.

Por esse motivo foi desenvolvido um script em Python que permite pegar uma imagem e criar novos exemplos (data augmentations), realizando alterações como: rotação, zoom, alteração de brilho e inversão da imagem, Figura 21.

**Figura 21 - Exemplo do Data Augmentations - imagem 1: alteração do brilho e inversão; imagem 2: alteração do brilho; imagem 3: alteração do brilho e zoom; imagem 4: alteração do brilho, zoom e inversão**



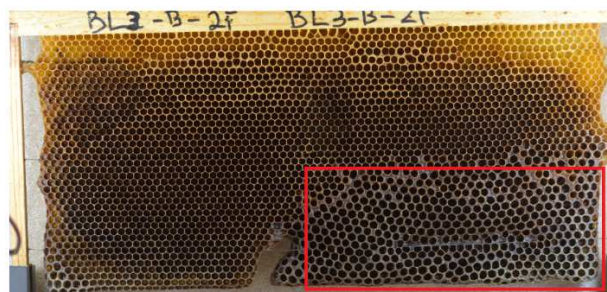
**Fonte: Autoria Própria (2022)**

### 3.3 Classe Zangão

Uma etapa do trabalho foi incluir a classe zangão na previsão no modelo, para isso a primeira etapa foi identificar com a base do DeepBee alvéolos de zangão. Para isso, foi utilizada a interface do DeepBee, para identificar manualmente as células de zangão e separar as demais classes.

Já que o software permite salvar os alvéolos dos quadros em diretórios separados por classe. Devido ao tamanho e espessura do zangão, seus alvéolos geralmente são maiores que os demais, essa característica contribui na tarefa de identificação de sua célula no quadro, conforme a Figura 22. As células de zangão foram rotuladas manualmente na classe “outros” e extraídas para o treinamento dos modelos.

**Figura 22 - Exemplos de imagens de alvéolos de zangão**

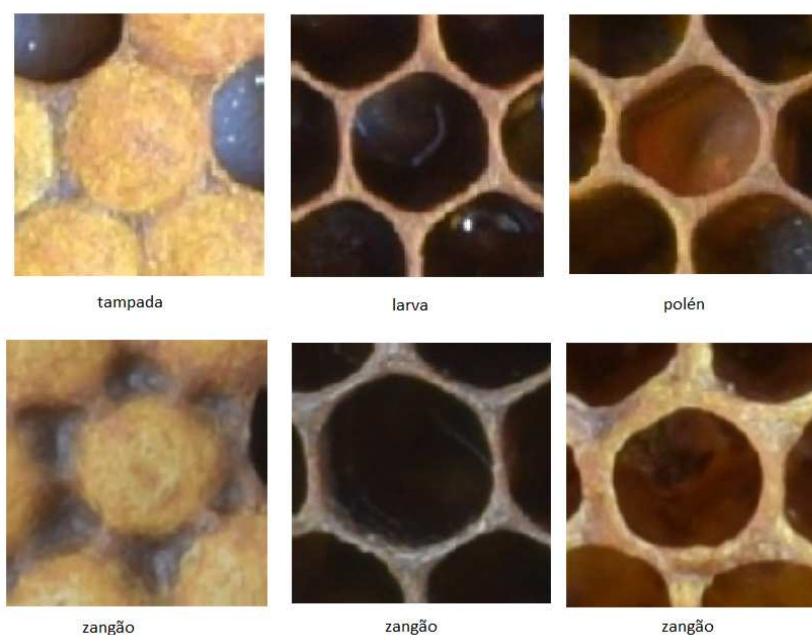


**Fonte: Autoria Própria (2022)**

Com a inclusão da classe zangão a próxima etapa foi realizar um treino com as demais classes para verificar o desempenho do modelo, tanto na ResNeXt

quanto na MobileNet. Durante o processo de treinamento, uma dificuldade foi encontrada, no processo de classificar a célula do zangão com as demais classes. O alvéolo do zangão apresentava semelhança entre outras classes, como pode ser observada na Figura 23.

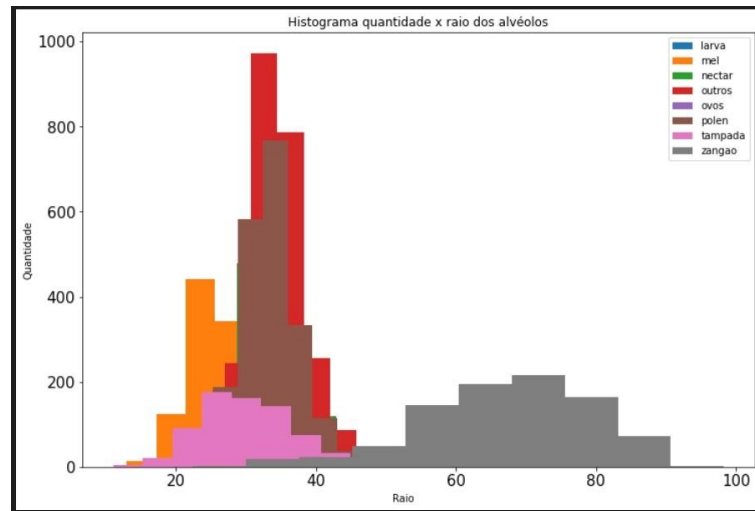
**Figura 23 - Semelhança de alvéolos do zangão com outras classes**



**Fonte: Autoria Própria (2022)**

Buscando contornar uma baixa assertividade dos modelos devido a semelhança das classes, foram desenvolvidos outras formas de classificar os alvéolos. No processo de identificar e contar os favos é aplicada a transformada de Hough, permitindo identificar cada círculo (alvéolo) e seu raio correspondente. Como o alvéolo do zangão tem maior dimensão, devido ao seu corpo, o raio do alvéolo do zangão é maior que as demais células, Figura 24.

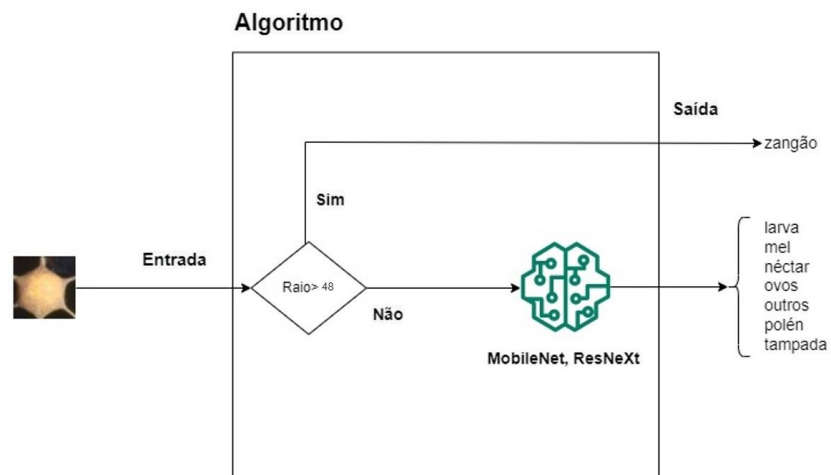
**Figura 24 - Diferença entre o raio dos alvéolos**



**Fonte: Autoria Própria (2022)**

Na Figura 24 é possível observar que a maioria dos raios dos alvéolos das outras classes ficam entre 20 e 40 pixels (px), um parâmetro (minRadius, maxRadius) do algoritmo de Hough, para identificar círculos. O alvéolo de zangão em sua maioria possui um raio superior a 48 px. Com essa informação, foi criado um algoritmo para uma etapa de pré-processamento, em que são identificados os alvéolos do zangão em relação ao tamanho do raio e os alvéolos que não são identificados como zangão, são aplicados os modelos, conforme a Figura 25.

**Figura 25 - Fluxograma do algoritmo**



**Fonte: Autoria Própria (2022)**

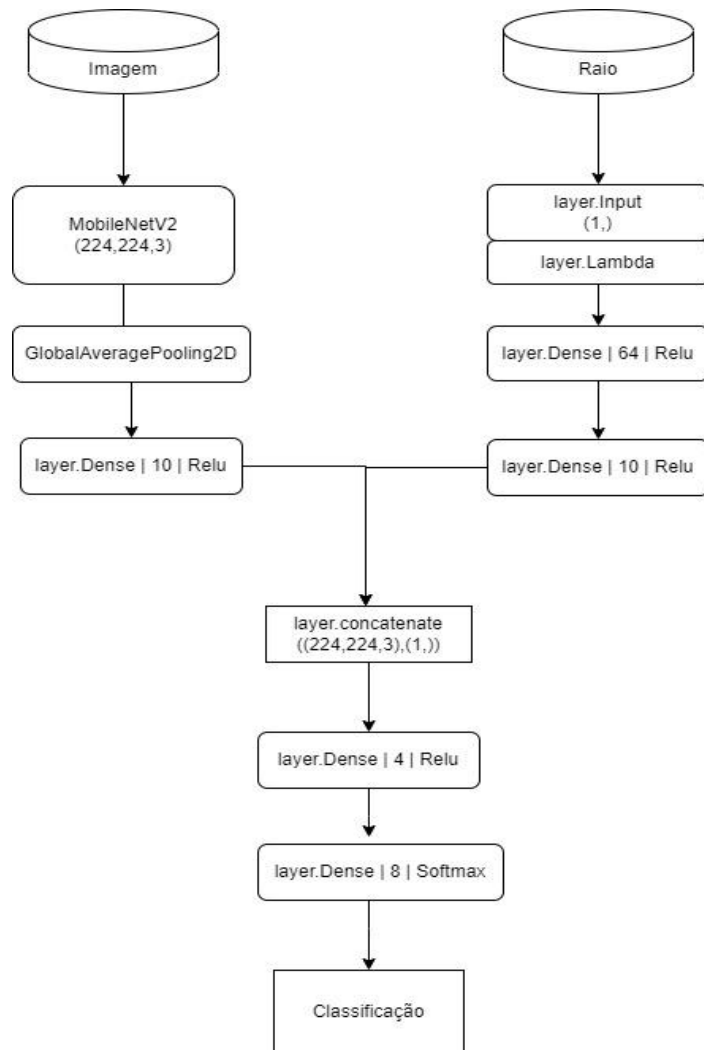
Como visto, os dois elementos que classificam bem os alvéolos são: o raio e sua foto, os dois são tipos de dados diferentes. Existe uma técnica que permite utilizar múltiplas entradas em uma rede neural, para isso é necessário alterar sua arquitetura (PYIMAGESEARCH, 2019).

Foi alterada a arquitetura da rede para receber os tipos de dados diferentes. Primeira etapa foi desenvolver uma rede *feedforward* para a entrada do tipo de dado do raio. Como apresentado no capítulo 2 do referencial teórico na subseção de rede neural, tipos de arquitetura de redes neurais, a *feedforward* é um tipo de arquitetura, as redes *feedforward* não possuem loop em sua estrutura, o fluxo de informação é sempre para frente (RICH; KNIGHT; NAIR, 1983).

A segunda etapa foi desenvolver a rede neural convolucional (MobileNet) que recebe como entrada a imagem do alvéolo. A forma que a rede recebe os dados são: (224,224,3) em que 224 x 224 px é referente a largura e altura da imagem, e o valor 3 devido a imagem ser colorida (três canais de cores). Com a inclusão do valor numérico do raio, a entrada da rede passa a ser:((224,224,3),(1,)).

Com as redes criadas a próxima etapa é concatenar as entradas e utilizar na última camada uma função de ativação softmax, como mostra a Figura 26. Com a estrutura criada, foi realizado o treinamento do modelo com a metodologia de aprendizado supervisionado, passando as imagens com os raios dos alvéolos, com sua classe correspondente.

**Figura 26 - Nova Arquitetura**

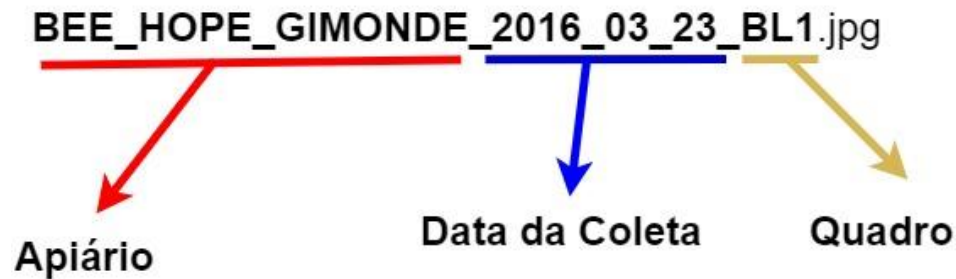


**Fonte: Autoria Própria (2022)**

### 3.4 Monitoramento

Um dos objetivos do projeto é desenvolver um método de acompanhamento de colmeias, permitindo uma análise da evolução por quadro, como surgimento de larvas, mel, pólen, etc. O protocolo da coleta de imagens do DeepBee, permite identificar a data da coleta, nome do apiário e nome quadro, através do nome da imagem, conforme a Figura 27.

Figura 27 - Formato do nome do arquivo



Fonte: Autoria Própria (2022)

Foi desenvolvido um código em python que coleta essas informações, e gera uma tabela com a data da coleta, apiário e o nome do quadro, por imagem, conforme a Figura 28.

Figura 28 - Formato em Tabela

Data Coleta	Apiário	Quadro
15/06/2016	BEE HOPE GIMONDE	BL1
18/08/2016	BEE HOPE GIMONDE	BL2
17/06/2016	BEE HOPE GIMONDE	BL2
28/07/2016	BEE HOPE GIMONDE	BL2
07/01/2016	BEE HOPE GIMONDE	BL5
20/06/2016	BEE HOPE GIMONDE	BL6
12/10/2016	BEE HOPE GIMONDE	BL3

Fonte: Autoria Própria (2022)

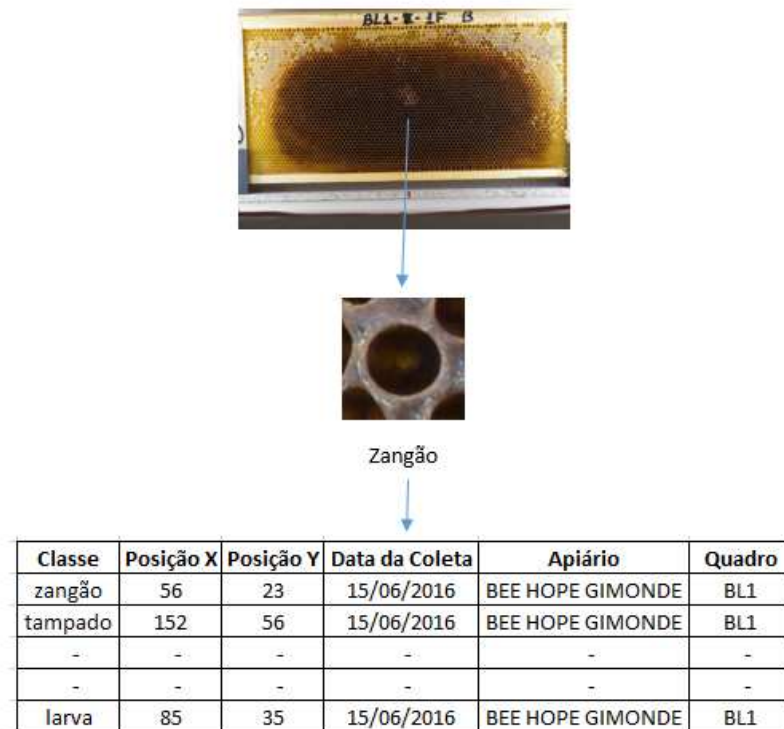
A próxima etapa foi encontrar informação por alvéolo, e agrupar por data da coleta, apiário e quadro. A primeira informação do alvéolo, é a sua classe identificada pelo modelo, a segunda informação é a posição do alvéolo identificada no quadro.

No mesmo código utilizado para formatar os dados de coleta, foram incluídas as informações dos alvéolos. Como um quadro pode ter milhares de alvéolos, o código gera um arquivo no formato csv, que é um arquivo de texto simples que armazena informações de planilhas e tabelas, cada imagem de quadro vai ter um arquivo em csv.

O arquivo tem o nome da classe do alvéolo, posição central do alvéolo (x,y), data de coleta do quadro, nome do apiário e nome do quadro. Portanto, além de ter

a informação do alvéolo, serão acrescentados os dados da coleta, conforme a Figura 29.

**Figura 29 - Exemplo do processo de monitoramento**



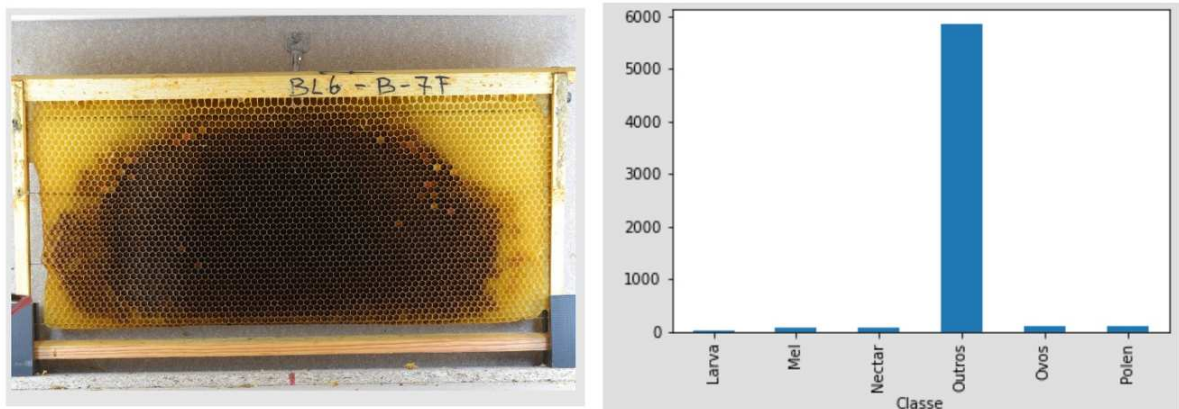
**Fonte: Autoria Própria (2022)**

Com o algoritmo criado e gerada a identificação dos alveolos por posição, a próxima etapa foi gerar gráficos com os dados. Como foram gerados arquivos no formato csv, foi desenvolvido um código em python, que realiza a leitura desses arquivos e gera gráficos seguindo duas formas diferente de análise.

Na primeira é analisado o quadro por inteiro, nessa etapa é contado o número de alveolos por classe, e gerados gráficos de linha, barra e posição do alvéolo. A ideia é acompanhar o desempenho por quadro, conforme a Figura 30, que apresenta o gráfico de barras.



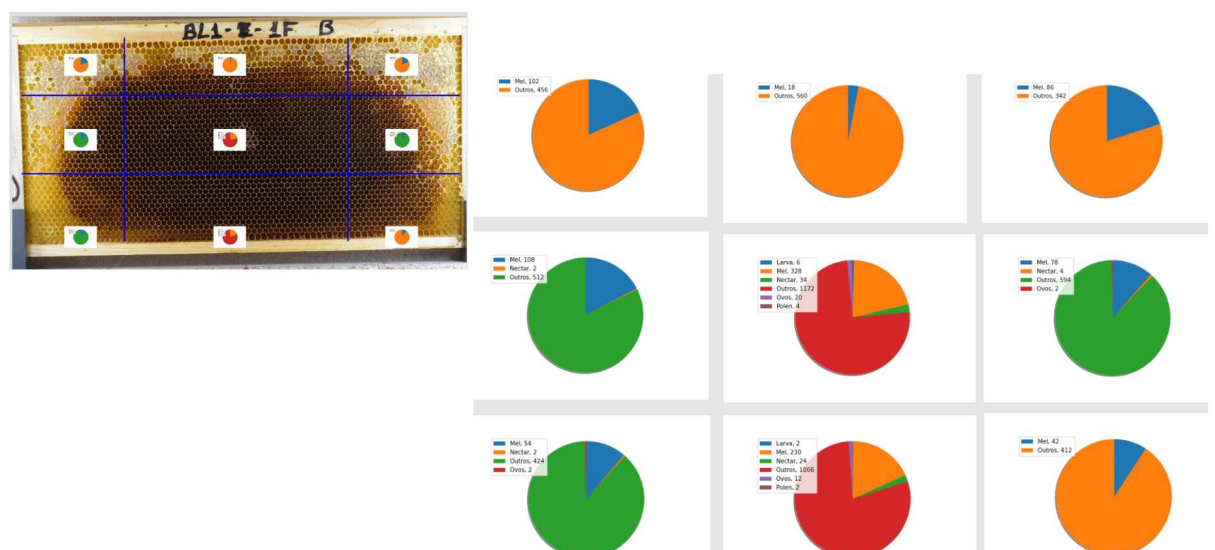
Figura 30 - Gráfico por linha



Fonte: Autoria Própria (2022)

Já na segunda forma de gerar gráficos, são utilizadas regiões do quadro para as análises. O quadro é dividido em nove partes, em que para cada parte é gerado um gráfico de pizza para contagem de alvéolos identificados. Essa forma de análise, permite verificar não apenas as classes que estão sendo identificadas e no período normal da colmeia, como também verificar se elas estão surgindo nos locais corretos do quadro, informação também importante, conforme a Figura 31.

Figura 31 - Gráfico por quadrante



Fonte: Autoria Própria (2022)

## 4 RESULTADOS

Nessa etapa será avaliado o desempenho dos modelos, pretende-se verificar se a ResNeXt possui melhor resultado que a MobileNet, modelo utilizado no DeepBee. Além de incluir a classe zangão e verificar o seu desempenho do modelo. O primeiro treinamento foi realizado utilizando a base do DeepBee sem a inclusão da classe zangão e sem data augmentation, como visto na etapa de aquisição de imagem. Foram utilizadas 10.783 imagens para avaliar os modelos, Figura 32, e para avaliar alguns resultados foram utilizados três métricas estáticas: acurácia, recall e f1-score.

**Figura 32 - Primeiro resultado do treino**



**Fonte: Autoria Própria (2022)**

A rede ResNeXt apresentou o mesmo valor de acurácia e f1-score, os dois modelos apresentaram um acerto de 97%, só seu desempenho entre classes apresentou uma divergência de resultados que podem ser visto na matriz de confusão da ResNeXt, conforme a Tabela 1.

**Tabela 1 - Resultado da Matriz - ResNeXt**

	Tampada	Outros	Ovos	Mel	Larva	Nectar	Polen
Tampada	<b>914</b>	0	0	1	2	0	0
Outros	5	<b>2573</b>	24	2	12	7	9
Ovos	0	53	<b>653</b>	0	20	2	0
Mel	1	1	0	<b>1289</b>	0	2	8
Larva	0	17	11	0	<b>1241</b>	2	3
Nectar	0	30	1	4	6	<b>1733</b>	5
Polen	0	35	1	0	7	8	<b>2101</b>

Fonte: Autorial Própria (2022)

Nela é possível identificar visualmente como está o acerto por classe do modelo, exemplo, a classe “tampado” no teste foram utilizadas 917 imagens o modelo classificou corretamente 914, errou 3 imagens apenas. A classe que o modelo mais errou foi a de “ovos”. Já em relação a MobileNet, conforme a Tabela 2, as classes apresentaram um melhor resultado, a pior classe do modelo foi “outros”.

**Tabela 2 - Resultado da Matriz - MobileNet**

	Tampada	Outros	Ovos	Mel	Larva	Nectar	Polen
Tampada	<b>914</b>	0	0	2	0	0	1
Outros	0	<b>2546</b>	41	1	12	12	20
Ovos	0	13	<b>712</b>	0	3	0	0
Mel	11	0	0	<b>1291</b>	0	1	8
Larva	0	10	26	0	<b>1229</b>	1	8
Nectar	1	19	2	3	2	<b>1744</b>	8
Polen	0	7	8	0	2	2	<b>2133</b>

Fonte: Autorial Própria (2022)

Com intuito de melhorar os resultados dos modelos, foi aplicado o data augmentation nos dados de treino, conforme apresentado na aquisição de imagem, além de aumentar a base de teste para 19.542 imagens (novas) as matrizes de confusão ficou conforme apresentado na Tabela 3 e 4.

**Tabela 3 - Resultado da Matriz - MobileNet - Data Augmentations**

	Tampada	Outros	Ovos	Mel	Larva	Nectar	Polen
Tampada	<b>2375</b>	0	0	2	0	0	1
Outros	0	<b>2546</b>	41	1	12	12	20
Ovos	1	36	<b>2142</b>	0	10	0	0
Mel	15	4	0	<b>2743</b>	0	2	11
Larva	4	18	37	0	<b>2666</b>	1	9
Nectar	5	30	15	4	9	<b>3168</b>	9
Polen	0	10	9	0	2	4	<b>3588</b>

Fonte: Autoria Própria (2022)

**Tabela 4 – Resultado da Matriz - ResNeXt - Data Augmentations**

	Tampada	Outros	Ovos	Mel	Larva	Nectar	Polen
Tampada	<b>2375</b>	0	0	3	0	0	0
Outros	0	<b>2482</b>	70	0	40	15	25
Ovos	1	37	<b>2115</b>	0	34	1	1
Mel	41	13	0	<b>2691</b>	1	1	8
Larva	0	8	21	0	<b>2700</b>	1	5
Nectar	5	25	23	6	17	<b>3146</b>	18
Polen	0	15	8	1	22	5	<b>3562</b>

Fonte: Autoria Própria (2022)

Os resultados dos dois modelos ficaram muito próximos, mas novamente a MobileNet apresentou um melhor resultado entre as classes e o valor de acurácia, recall e f1-score também, conforme a Figura 33.

**Figura 33 - Resultado dos Modelos - com data augmentation**

Fonte: Autoria Própria (2022)

#### 4.1 Treinamento com a classe zangão

Nessa etapa pretende-se incluir mais uma classe, os alvéolos de zangão, coletadas conforme citadas na etapa de aquisição. Foram identificadas 33.223 imagens da classe zangão e separadas 32.111 imagens para o treino e 1.112 para o teste. A primeira arquitetura treinada com a classe zangão foi a MobileNet, realizado um primeiro treino e avaliado o resultado Figura 34.

O modelo apresentou uma baixa taxa de acerto, com f1-score de 50%, foi necessário realizar um novo treino aumentando o número de interações (épocas) no modelo e diminuir a taxa de aprendizagem para 0,0001 (treino 2), buscando uma melhor taxa de acerto.

**Figura 34 – Resultado da MobileNet com a classe zangão**



**Fonte: Autoria Própria (2022)**

O modelo apresentou um melhor resultado com f1-score de 63%, mas inferior aos 98% de taxa de acerto da MobileNet sem a classe zangão. O primeiro treino com a ResNeXt e a inclusão da classe zangão, apresentou acurácia de 20%, *recall* de 19% e f1-score 16%, um resultado abaixo do esperado. Aumentando o número de interações (épocas) e diminuindo a taxa de aprendizado (treino 2), apresentou o resultado, Figura 35. O melhor resultado da ResNeXt foi f1-score: 39%, inferior aos 97% de taxa de acerto sem o zangão.

**Figura 35 – Resultado da ResNext com a classe zangão**



**Fonte: Autoria Própria (2022)**

Os dois modelos tiveram resultados bem inferiores em relação taxa de acerto, com a inclusão da classe zangão. A dificuldade de classificar a célula do zangão com as demais, é devido à semelhança entre elas, como visto no capítulo 3. Devido a diferença entre o raio do alvéolo do zangão em relação as demais, foi criado um algoritmo para verificar o melhor limiar em relação ao raio para identificar a classe zangão e as demais, nele é calculado o valor de acurácia, recall e f1-score do modelo com a mudança do raio, conforme a Figura 36.

**Figura 36 – Métricas em relação ao raio**

<b>Raio</b>	<b>Acurácia</b>	<b>Recall</b>	<b>f1-score</b>
32	0.420171	0.475666	0.499910
36	0.761048	0.783486	0.792696
40	0.911618	0.918517	0.907138
44	0.960175	0.961492	0.953976
<b>48</b>	<b>0.972202</b>	<b>0.971507</b>	<b>0.968111</b>
52	0.972113	0.969940	0.967909
56	0.967659	0.962646	0.962160
60	0.960264	0.950936	0.952052

**Fonte: Autoria Própria (2022)**

Com essa nova estrutura foi realizada uma nova avaliação com a MobileNet, e como critério o raio acima de 48 para classificar a classe Zangão, que apresentou acurácia: 97%, *recall*: 97% e f1-score: 96%. Com a nova estrutura foi possível

classificar o zangão com as demais classes, com uma taxa de acerto de 97%, e a classe com menor desempenho foi acima de 94%. A matriz de confusão da MobileNet, conforme Tabela 5.

Já o modelo com a inclusão do raio e a imagem como, atributos de entrada da rede neural, apresentou uma acurácia de 95%. Foram avaliadas 500 imagens por alvéolo, a classe com melhor resultado foi a do zangão com 100% de taxa de acerto e a classe com pior resultado foi a polén, com acurácia de 85%. O modelo apresentou uma taxa de acerto acima de 85% por classe, como pode ser visto na matriz de confusão, conforme Tabela 6.

**Tabela 5 – Matriz de confusão da MobileNet com zangão**

	Tampada	Outros	Ovos	Mel	Larva	Nectar	Polen	Zangão
Tampada	<b>96.5%</b>	0%	0%	0.8%	0%	0.1%	0%	3.2%
Outros	0%	<b>94.6%</b>	1.8%	0%	0.7%	1.0%	0.3%	0.3%
Ovos	0%	1.6%	<b>97%</b>	0%	2%	0.1%	0.4%	0%
Mel	0.3%	0%	0%	<b>97.6%</b>	0%	0.2%	0%	0.2%
Larva	0%	0.4%	0.4%	0%	<b>94.3%</b>	0.1%	0.1%	0.3%
Nectar	0%	0.4%	0%	0.1%	0.1%	<b>95.7%</b>	0.1%	0%
Polen	0.1%	0.8%	0%	0.6	0.5%	0.4%	<b>97.5%</b>	0%
Zangão	3.0%	2.1%	0.8%	1.0%	2.4%	2.4%	1.6%	<b>95.9%</b>

Fonte: Autoria Própria (2022)

**Tabela 6 – Matriz de confusão da MobileNet, com o raio e a imagem do alvéolo**

	Larva	Mel	Nectar	Outros	Ovos	Polen	Tampado	Zangão
Larva	<b>93.8%</b>	0%	0.1%	2%	4%	0.1%	0%	0%
Mel	0%	<b>99.9%</b>	0%	0.001%	0%	0%	0%	0%
Nectar	0.01%	0%	<b>96.8%</b>	3%	0.19%	0%	0%	0%
Outros	0%	0.2%	0.3%	<b>94.2%</b>	5%	0.3%	0%	0%
Ovos	0.4%	0%	0%	3%	<b>96.6%</b>	0%	0%	0%
Polen	5.6%	1%	1%	4%	3%	<b>85.4%</b>	0.1%	0%
Tampado	0%	3.8%	0%	0%	0%	0%	<b>96.2%</b>	0%
Zangão	0%	0%	0%	0%	0%	0%	0%	<b>100%</b>

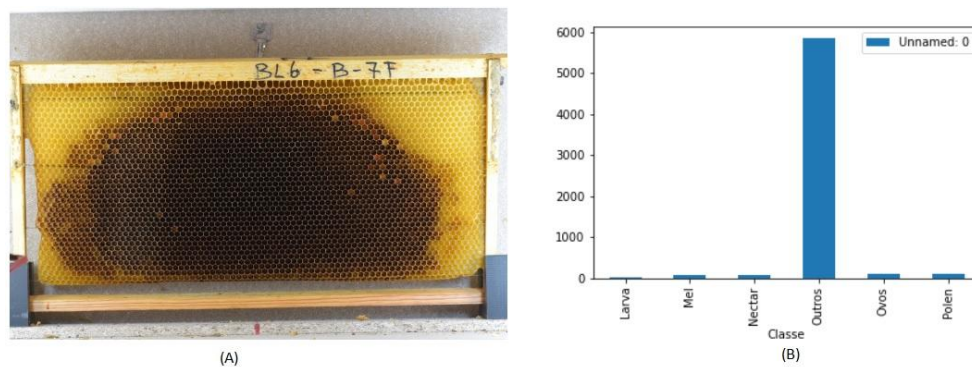
Fonte: Autoria Própria (2022)

## 4.2 Monitoramento

Com o algoritmo de geração de arquivos, com informação da colmeia criado, a próxima etapa foi gerar gráficos com os dados que permite auxiliar o apicultor no

acesso à informação. Um exemplo do monitoramento é a Figura 37 - (a), que foi coletada no apiário Bee Hope Gimonde no dia 30 de abril de 2016, do quadro BL6. Aplicado o modelo e o algoritmo de monitoramento, o primeiro gráfico gerado é de contagem de alvéolos classificados e suas classes, Figura 37 - (b). Na figura é apresentado um quadro vazio, no início da colmeia, devido aos alvéolos vazios, a classe mais identificada é a "outros".

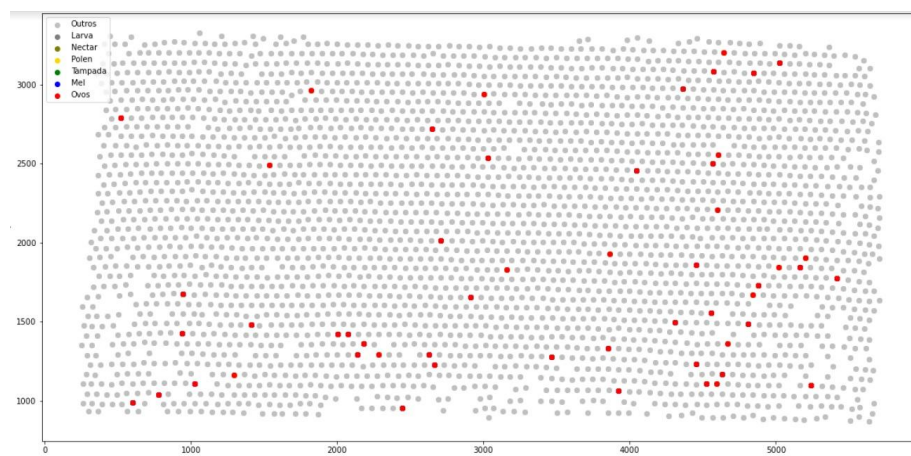
**Figura 37 – Gráfico de contagem de classes**



Fonte: Autoria Própria (2022)

O algoritmo gera um gráfico por posição das classes identificadas separadas por cor, a ideia é ajudar o apicultor não só a saber a quantidade identificada, mas também ter um histórico de posição, Figura 38.

**Figura 38 – Gráfico de posição das classes**

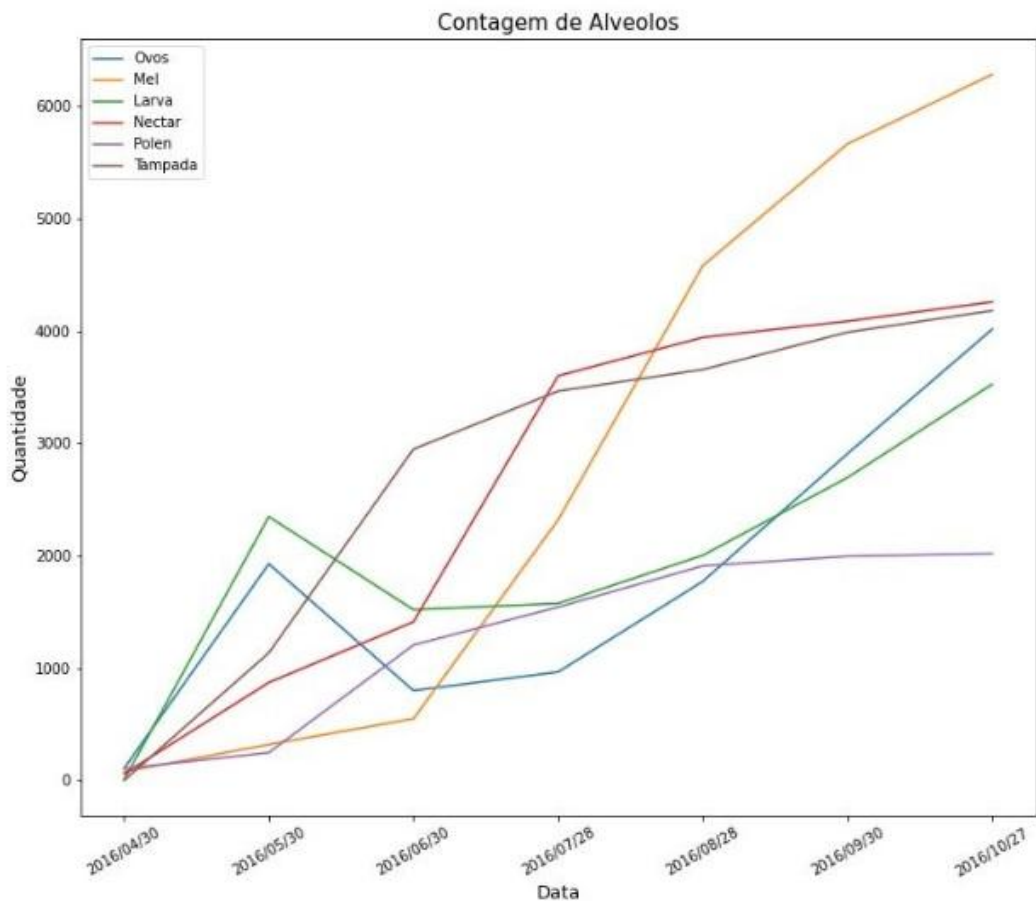


Fonte: Autoria Própria (2022)



Outro gráfico gerado utilizando a imagem do quadro inteiro, é o gráfico de linhas, a ideia principal dele é acompanhar a evolução dos alvéolos em um quadro, em uma linha de tempo, conforme a Figura 39. Nela é apresentado o mesmo quadro, avaliado em sete meses com fotos tiradas a cada 30 dias, Figura 39. Com intuito de determinar se está ocorrendo o comportamento normal do quadro e também comparar com os outros da mesma colmeia.

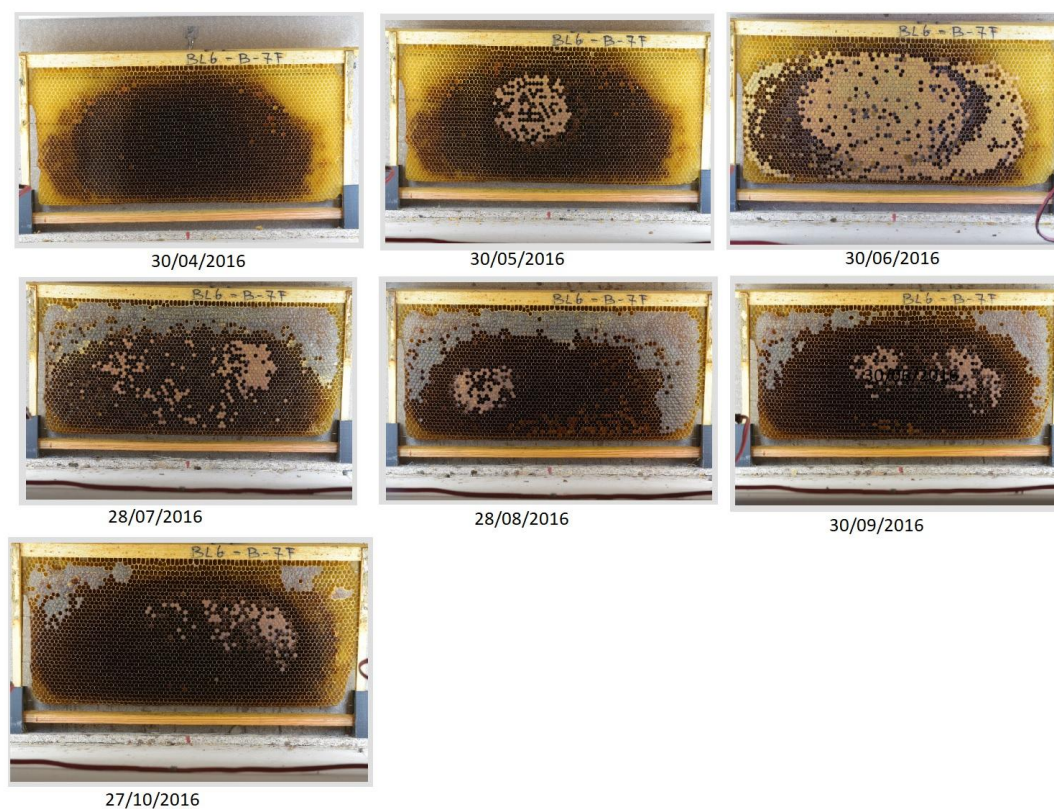
**Figura 39 – Gráfico de linhas do quadro**



**Fonte: Autoria Própria (2022)**

Na Figura 40 é apresentado o início da colmeia com alvéolos vazios, a mudança após um mês com o surgimento dos primeiros ovos, que viram pupa e depois larvas, e nos meses finais gerando alvéolos de armazenamento de mel.

**Figura 40 – Evolução de um quadro por 7 meses**

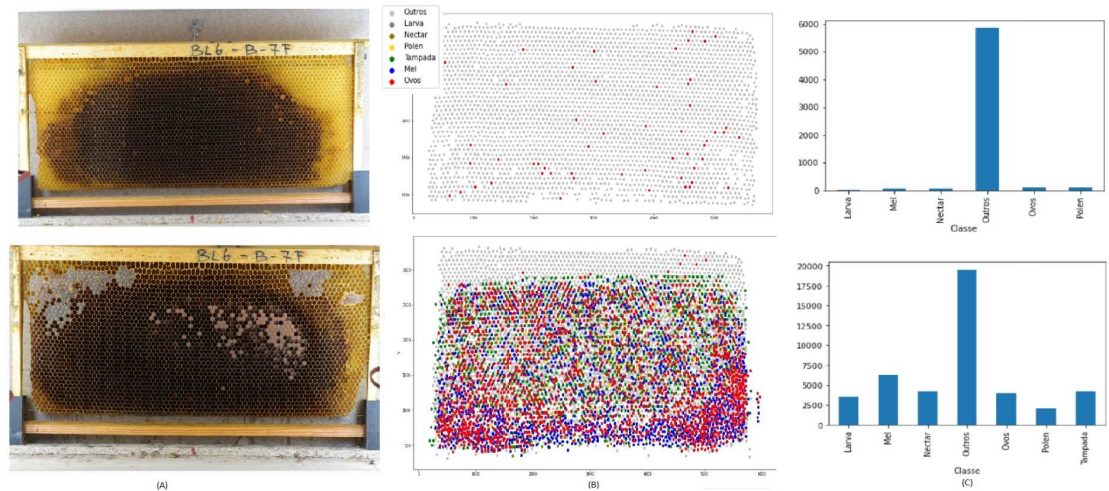


**Fonte: Autoria Própria (2022)**

Para exemplificar a evolução de um quadro, o mesmo algoritmo dos gráficos de posição e barra, foi aplicado no mesmo quadro (BL6), só que seis meses depois, 27 de outubro de 2016, Figura 41 - (a). Na Figura 41 - (b,c) estão as classes identificadas e por posição, mostra o surgimento de classes como o mel e o néctar.

A ideia do algoritmo é guardar um histórico mês a mês, com as informações salvas em um arquivo 'nome\_imagem.csv' e imagens com gráficos por contagem de classe, posição e linha do tempo, que permita um acesso mais fácil para o apicultor e auxiliar o monitoramento e tomadas de decisões em uma colmeia.

**Figura 41 – Gráficos seis meses depois**

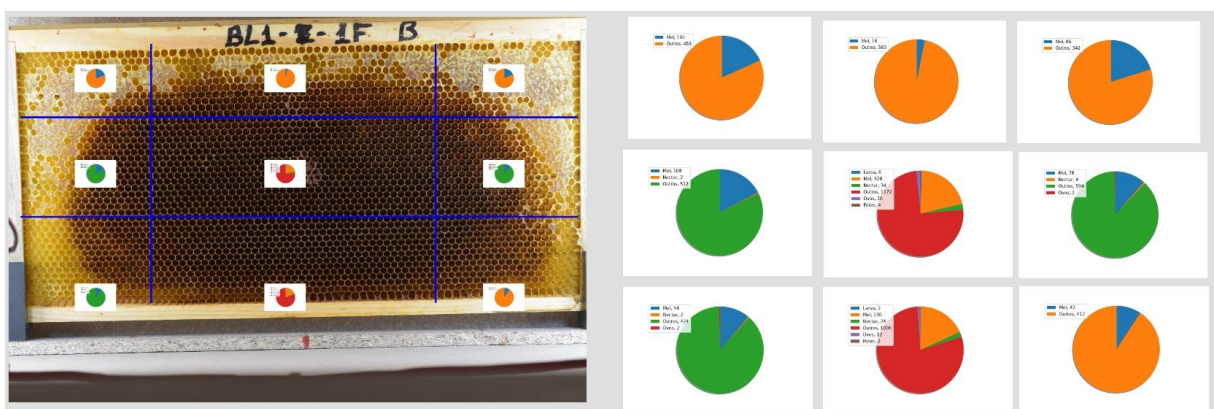


Fonte: Autoria Própria (2022)

Com a identificação da classe e a posição do alvéolo, foi possível também gerar gráficos por quadrante no quadro. Nessa visão é possível acompanhar o quadro por regiões, facilitando a identificação de anomalias, já que geralmente doenças no quadro surgem no centro e o mel nas bordas superiores.

No algoritmo o quadro é dividido em nove partes, sendo gerados contagens de classes identificadas pelo modelo por região, e ainda são gerados gráficos de regiões separado da imagem, conforme a Figura 42.

**Figura 42 – Gráficos por região no quadro**

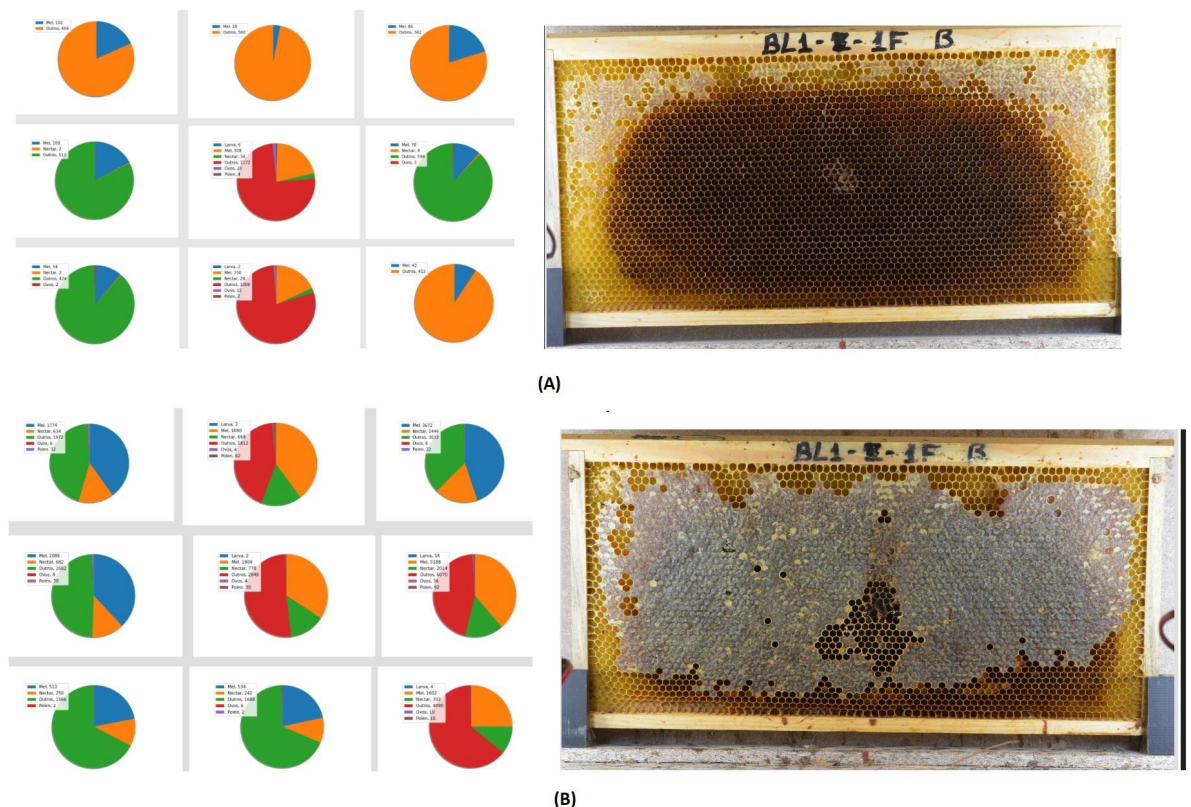


Fonte: Autoria Própria (2022)



Para exemplificar a evolução de um quadro por região, foi aplicado o algoritmo no quadro BL1 em 23 de março de 2016 e aplicado novamente no mesmo quadro sete meses depois, 27 de outubro de 2016, pode ser visto na Figura 43 que o número de classes por região aumentaram, com maior quantidade para classe mel.

**Figura 43 – Gráficos por região no quadro - Figura (A) quadro BL1 de 23 de março de 2016 com poucas classes identificadas e baixa quantidade alvéolos por classe, e Figura (B) do mesmo quadro coletada em 27 de outubro de 2016, com mais classes identificadas por região, com destaque para classe mel**

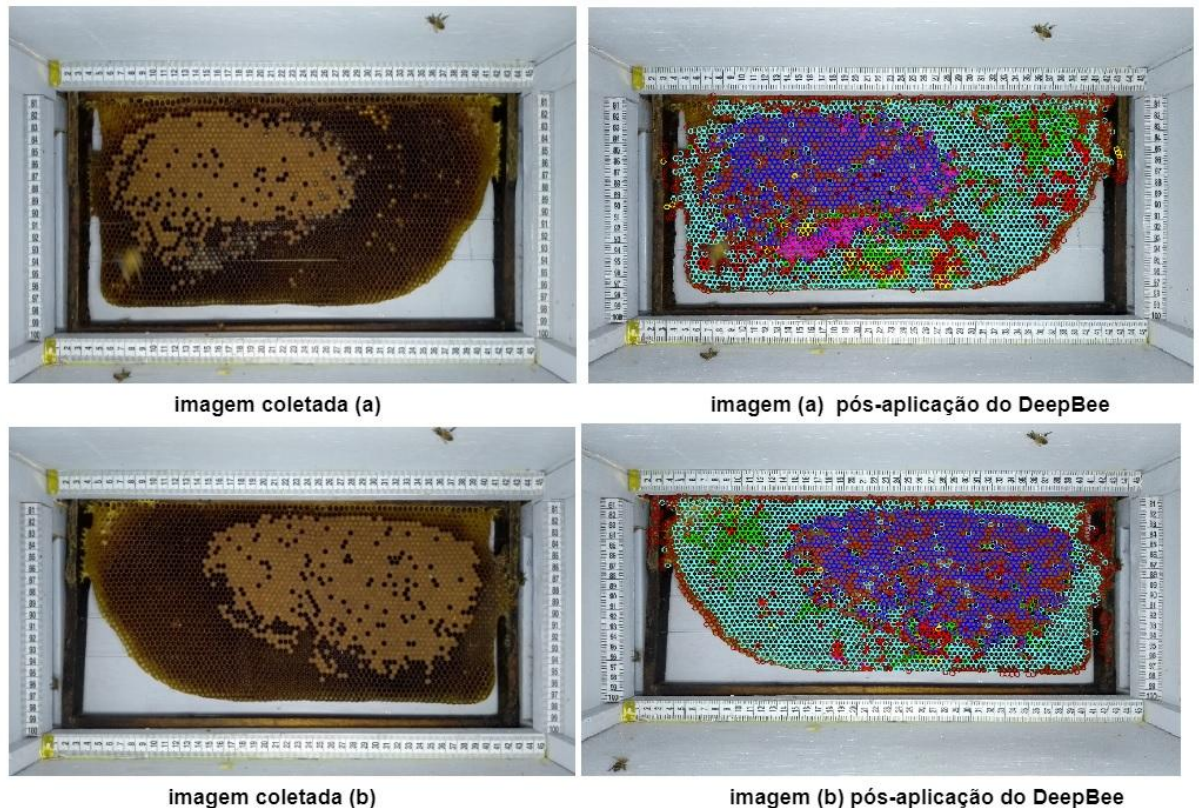


Fonte: Autoria Própria (2022)

Outro experimento realizado no projeto, foi utilizar imagens de quadro de colmeias nacionais, com abelhas africanizadas. O propósito desse teste foi verificar a capacidade do DeepBee identificar alvéolos em colmeias diferentes da europeia e o seu desempenho detectar e classificar os alvéolos em um protocolo de coleta diferente do DeepBee. Foram utilizadas 20 imagens da Empresa de Pesquisa

Agropecuária e Extensão de Santa Catarina, Epagri, o DeepBee não só conseguiu identificar os alvéolos como também classificá-los, conforme a Figura 44.

**Figura 44 – Imagens de quadros coletados no Brasil, sendo aplicado o DeepBee. Mesmo com um protocolo diferente, o modelo conseguiu detectar os alvéolos e os classificar**



**Fonte: Autoria Própria (2022)**

## 5 CONCLUSÃO

O principal objetivo desse trabalho, foi aprimorar o DeepBee com novos estudos e técnicas, esse trabalho se dividiu em três etapas principais: testar novas arquiteturas de redes neurais convolucionais, aprimorar o DeepBee para identificação dos alvéolos de zangão e criar algoritmos para o monitoramento de colônias. Antes de testar novas arquiteturas, foi necessário um estudo do software DeepBee visando verificar seu funcionamento e desempenho.

O software utiliza a arquitetura MobileNet para fazer as sugestões dos alvéolos, com isso a primeira etapa foi avaliar modelos não testados no

desenvolvimento do software para ser comparado com a MobileNet. O modelo escolhido foi a ResNeXt, uma arquitetura que além de apresentar bons resultados, utiliza uma biblioteca (Torch) de aprendizagem de máquina diferente da MobileNet.

Foram criadas as duas redes com suas bibliotecas correspondentes, e utilizado a base do DeepBee para se fazer o treinamento. Foram realizados dois treinos com os modelos, um utilizando o data augmentation e outro sem, a primeira conclusão encontrada foi que utilizando mais exemplos no treino a MobileNet tem melhor desempenho com f1-score de 98%, anteriormente apresentava 97%.

Outra conclusão desse teste, foi que a classe 'outros' apresentou o pior resultado em ambos os modelos, fazendo sentido referente a semelhança a outras classes. Mesmo utilizando uma rede mais recente e aplicando técnicas (data augmentation) de melhora de treino, os modelos apresentaram uma taxa de acerto semelhante, mas a MobileNet apresentou um melhor resultado com f1-score de 98%.

Já a ResNeXt com uma f1-score de 97%, poderia ser utilizada como complemento na sugestão da MobileNet, já que apresentou resultado semelhante entre as classes e superior na identificação do alvéolo de larva.

Na etapa de incluir a classe do zangão no modelo, foi utilizado o software do DeepBee para separar os alvéolos do zangão dos demais, gerando 33.223 imagens de alvéolo de zangão, foram realizados dois treinos com cada um dos modelos (ResNeXt e MobileNet). Os dois apresentaram um resultado com baixa taxa de acerto f1-score 40%, a primeira conclusão desse teste foi que a classe zangão era muito parecida com outras classes, dificultando sua identificação.

Para treinar com a classe zangão, foi necessário um maior entendimento de seu alvéolo, foram realizadas análises e avaliadas algumas técnicas, todas envolvendo o raio do alvéolo.

A que obteve melhor resultado foi uma rede que processava tanto as imagens dos alvéolos como também a informação do raio, para isso foi desenvolvido uma rede neural que aceitava dois tipos de entrada. O modelo do zangão apresentou uma f1-score de 95% e uma alta taxa de acerto entre as classes, sendo que para a classe zangão o modelo acertou 100%, possibilitando a identificação de uma nova classe no DeepBee.

O último objetivo do trabalho era facilitar o monitoramento da colmeia, foram desenvolvidos algoritmos que permitiam visualizar e quantificar o desempenho de

uma colmeia durante um período. Com os gráficos criados é possível verificar as mudanças em cada quadro na colmeia por regiões do quadro.

Todos os objetivos específicos traçados no início do projeto, foram abordados e geraram artefatos que poderão ser utilizados no DeepBee. Portanto, objetivo de trazer melhorias para ferramenta (como a inclusão da classe zangão) foi desenvolvido. A ferramenta se mostrou muito eficiente até com modelos atuais, mesmo assim foi importante realizar comparativos entre elas, já que podem ser utilizadas juntas e melhorar a ferramenta.

Os próximos passos seriam incluir ou melhorar esses artefatos na ferramenta, um exemplo seria não só identificar a classe zangão, separar entre tampada, larva e ovos de zangão. Outro projeto futuro seria utilizar bases de imagens nacionais, para identificar e classificar alvéolos. Durante a fase final do projeto foi realizado um teste com imagens de abelhas africanizadas, e mesmo com uma colmeia diferente da europeia e um protocolo de coleta bem diferente do DeepBee, a ferramenta conseguiu identificar os alvéolos e classificá-los.

## REFERÊNCIAS

- ALVES, T. S.; et al. Automatic detection and classification of honey bee comb cells using deep learning. **Computers and electronics in agriculture**, v. 170, n. 12, p. 5–31, 2020.
- BARBOSA, A. **Criação de abelhas**. Master Editora, v. 1, p. 1–60, 2008.
- BERKAYA, S. K.; SORA, E. G.; GUNAL, S. Deep learning-based classification models for beehive monitoring. **Ecological informatics**, v. 64, n. 12, p. 10–20, 2017.
- BEZERRA, A. L. D. Acoes terapeuticas da geleia real. **Revista brasileira de educação e saúde - REBES**, v. 1, p. 1–10, 2018.
- BJERGE, K. A computer vision system to monitor the infestation level of varroa destructor in a honeybee colony. **Computers and electronics in agriculture**, v. 164, n. 12, p. 10–20, 2019.
- BOZEK, K. Markerless tracking of na entire honey bee colony. **Nature communications**, v. 13, n. 12, p. 10–20, 2021.
- CAMPOS, L. A.; PERUQUETTI, L. C. Biologia e criação de abelhas sem ferrão. **Embrapa**, v. 1, p. 1–60, 2008.
- CARVALHO, S. L. **Inteligencia artificial - uma abordagem de aprendizado de maquina**. São Paulo: LTC, 2021.
- LUSTOSA, S. R.; GALINDO A. B.; NUNES, L. C. **Própolis: atualizações sobre a química e a farmacologia**. Scielo Brasil, v. 1, p. 5–10, 2008.
- COLIN, T. The development of honey bee colonies assessed using a new semi-automated brood counting method: Combcount. **PloS ONE**, v. 13, n. 3, p. 10–20, 2018.
- CPT. **Abelhas com ferrão: origem da apis mellifera e mecanismo de defesa**. 2019. Disponível em: <https://www.cpt.com.br/artigos/abelhas-com-ferrao-origem-da-apis-mellifera-e-mecanismo-de-defesa> . Acesso em: 28 maio 2021.
- CRANE, E. **The world history of beekeeping and honey hunting**. São Paulo: Routledge, 1999.
- CAMARGO, R. C. R. **Produção de mel**. EMBRAPA, v. 1, p. 12–17, 2002.



DJEKOUNE, A. O.; MESSAOUDI, K.; AMARA, K. Incremental circle hough transform: Na improved method for circle detection. **Optik**, v. 133, n. 12, p. 17–31, 2017.

ESCOBAR A. L. S.; XAVIER, F. B. **Propriedades fitoterápicas do mel de abelhas**. Master Editora, v. 1, p. 1–8, 2008.

FACELI, K. **Inteligencia artificial uma abordagem de aprendizado de maquina**. Rio de Janeiro: Brasport, 2011.

FALK, D. **The turing test measures something, but It's not intelligence**. 2014. Disponível em: <https://www.smithsonianmag.com/innovation/turing-test-measures-something-but-not-intelligence-180951702/>. Acesso em: 22 fev. 2023.

FERNANDES, T. **A importância das abelhas na vida do planeta**. 2009. Disponível em: <https://www.rondoniagora.com/artigos/a-importancia-das-abelhas-na-vida-do-planeta-por-tadeu-fernandes>. Acesso em: 26 maio 2021.

FILHO, O. M.; NETO, H. V. **Processamento digital de imagens**. Rio de Janeiro: Brasport, 1999.

GOMIDE, J. V. B. **Imagem digital aplicada**. Rio de Janeiro: Elsevier, 2014.

GONZALEZ, R. C.; WOODS, R. E. **Digital image processing**, Global Edition. Rio de Janeiro: Pearson, 2018.

HARRISON, M. **Machine learning – guia de referência rápida: trabalhando com dados estruturados em Python**. São Paulo: novatec, 2019.

HAYKIN, S. **Redes neurais principios e pratica**. São Paulo: bookman, 2001.

HOWARD, A. G. **Mobilenets: efficient convolucional neural networks for mobile vision applications**. 2015. Disponível em: <https://arxiv.org/pdf/1704.04861.pdf>. Acesso em: 31 jul. 2021.

IBGE. **Produção de mel de abelha**. 2019. Disponível em: <https://www.ibge.gov.br/explica/producao-agropecuaria/mel-de-abelha/br>. Acesso em: 27 maio 2021.

JENYNS, G. F. **A Book about bees - their history, habits, and Instincts**. São Paulo: Home Farm Books, 2016.

KUTUK, Y. **Activation functions: activation functions in deep learning with latex applications**. Eua: Peter Lang, 2022.

NART, C. D. Image recognition using convolutional neural networks for classification of honey bee subspecies. **Apidologie**, v. 1, p. 1–5, 2022.

NASCIMENTO J. E. M.; PEDROSO, C. G. S. J.; SOUZA, T. H. S.; TOLEDO, V. A. A. **O polen apícola e seus benefícios a saúde humana**. UEM, v. 1, p. 1–10, 2018.

NGO, T. N. Automated monitoring and analyses of honey bee pollen foraging behavior using a deep learning-based imaging system. **Computers and electronics in agriculture**, v. 187, n. 12, p. 10–20, 2021.

NORVIG, P.; RUSSELL, S. **Inteligencia artificial. 2a ed.** Rio de Janeiro: Campus, 2004.

NORVIG, R. **Artificial intelligence: a modern approach | fourth edition| by pearson**. Eua: Pearson, 2021.

NUMPYDL. **Convolution neural networks**. Disponível em: <https://numpydl.readthedocs.io/en/0.2.0/tutorials/CNN>. Acesso em: 04 mar. 2018.

NUNES, L.; CORREIA, M.; SILVEIRA, E. A. Produção de cera. **Produtor rural**, v. 52, p. 5–10, 2018.

PYIMAGESEARCH. **Keras: multiple inputs and mixed data**. Disponível em: <https://pyimagesearch.com/2019/02/04/keras-multiple-inputs-and-mixed-data/>. Acesso em: 02 ago. 2022.

REQUIER, F.; ANTUNEZ, D. G. Trends in beekeeping and honey bee colony losses in latin america. **Journal of apicultural research**, v. 57, n. 3, p. 657–662, 2018.

REUTERS. **Coronavírus afeta produção de mel da China, maior produtora mundial**. Disponível em: [https://www.noticiasagricolas.com.br/noticias/politica-economia/253016-coronavirus-afeta-producao-de-mel-da-china-maior-produtora-mundial.html#YK7\\_ZKhKg2y](https://www.noticiasagricolas.com.br/noticias/politica-economia/253016-coronavirus-afeta-producao-de-mel-da-china-maior-produtora-mundial.html#YK7_ZKhKg2y). Acesso em: 26 maio 2021.

RICH, E.; KNIGHT, K.; NAIR, S. B. **Artificial Intelligence**. Rio de Janeiro: McGraw-Hill, 1983.

ROCHA, S. J. **Apicultura. manejo de alta produtividade**. São Paulo: Agrolivros, 2018.

SERVICO NACIONAL DE APRENDIZAGEM RURAL. **Manejo de apiário para produção de mel: Senar**. Brasil, 2009. 20-30 p.

SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. **Redes neurais artificiais para engenharia e ciencias aplicadas, fundamentos teoricos e aspectos praticos**. São Paulo: Artliber, 2015.

SMITH. **Hough circle transform**. 2018. Disponível em: <https://imagej.net/plugins/hough-circle-transform>. Acesso em: 24 fev. 2023.

SNODGRASS, R. E. **The anatomy of the honey bee**. Inglaterra: Home Farm Books, 2016.

SOUZA, C. R. **Apitoxina: utilização do veneno da abelha apis mellifera**. pubvet, v. 1, p. 1–5, 2020.

TRADE. **The EU honey market slipped back slightly to \$1.4B**. 2020. Disponível em: <https://www.globaltrademag.com/the-eu-honey-market-slipped-back-slightly-to-1-4b>. Acesso em: 26 maio 2021.

VIDAL, M. **Evolução da produção de mel na área de atuação do bnb**. Caderno Setorial ETENE, v. 1, p. 1–5, 2020.

WIESE, S. H. J. **Nova apicultura**. Agro, v. 1, p. 1–60, 2020.

XIE, S. **Aggregated residual transformations for deep neural networks**. 2018. Disponível em: <https://arxiv.org/pdf/1611.05431.pdf>. Acesso em: 31 jul. 2021.

YOO, J. A machine learning based approach to study morphological features of bees. **International electronic conference on entomology**, v. 25, n. 3, p. 100–120, 2021.

ZANOTTA, D. C.; FERREIRA, M. P.; ZORTEA, M. **Processamento de imagens de satélite**. Rio de Janeiro: Oficina de Textos, 2019.