

Juliana Karl Araujo

**Sistema de Visão Computacional para
Determinação da Taxa de Crescimento e Previsão
de Colheita no Cultivo de Cogumelos Comestíveis**

Campos dos Goytacazes

2019

Juliana Karl Araujo

**Sistema de Visão Computacional para Determinação da Taxa
de Crescimento e Previsão de Colheita no Cultivo de
Cogumelos Comestíveis**

Dissertação apresentada ao Curso de Mestrado em Sistemas Aplicados à Engenharia e Gestão do Instituto Federal Fluminense como requisito parcial para obtenção do Grau de Mestre em Sistemas Aplicados à Engenharia e Gestão. Área de concentração: visão computacional. Linha de Pesquisa: engenharia.

Instituto Federal de Educação, Ciência e Tecnologia Fluminense – IFF
Mestrado Profissional em Sistemas Aplicados à Engenharia e Gestão
Programa de Pós-Graduação

Orientador: Prof. D.Sc. Rogério Atem de Carvalho

Coorientador: Prof. D.Sc. William da Silva Vianna

Campos dos Goytacazes

2019

Biblioteca Anton Dakitsch
CIP - Catalogação na Publicação

A663s Araujo, Juliana Karl
Sistema de Visão Computacional para Determinação da Taxa de
Crescimento e Previsão de Colheita no Cultivo de Cogumelos Comestíveis /
Juliana Karl Araujo - 2019.
69 f.: il. color.

Orientador: Rogério Atem de Carvalho
Coorientador: William da Silva Vianna

Dissertação (mestrado) -- Instituto Federal de Educação, Ciência e
Tecnologia Fluminense, Campus Campos Centro, Curso de Mestrado
Profissional em Sistemas Aplicados à Engenharia e Gestão, Campos dos
Goytacazes, RJ, 2019.
Referências: f. 44 a 48.


1. Visão Computacional. 2. Cultivo de Cogumelos. 3. Python OpenCV.
4. NodeRED. I. Atem de Carvalho, Rogério, orient. II. da Silva Vianna,
William, coorient. III. Título.

Juliana Karl Araujo

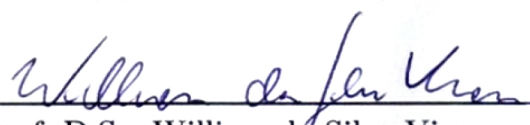
Sistema de Visão Computacional para determinação da Taxa de Crescimento e Previsão de Colheita no Cultivo de Cogumelos Comestíveis

Dissertação apresentada ao Curso de Mestrado em Sistemas Aplicados à Engenharia e Gestão do Instituto Federal Fluminense como requisito parcial para obtenção do Grau de Mestre em Sistemas Aplicados à Engenharia e Gestão. Área de concentração: visão computacional. Linha de Pesquisa: engenharia.


Trabalho aprovado. Campos dos Goytacazes, 19 de dezembro de 2019 :



Prof. D.Sc. Rogério Atem de Carvalho
Orientador
Instituto Federal Fluminense – IFF



Prof. D.Sc. William da Silva Vianna
Coorientador
Instituto Federal Fluminense – IFF



Prof. D.Sc. Renato Gomes Sobral
Barcellos
Instituto Federal Fluminense – IFF

Campos dos Goytacazes

2019

Dedico este trabalho a todos os professores que contribuíram para a minha formação profissional e me inspiraram de alguma maneira a buscar o conhecimento.

AGRADECIMENTOS

Gostaria de aproveitar este espaço para agradecer a todos aqueles que contribuíram de alguma forma para a realização deste trabalho.

Primeiramente à minha família que sempre me apoiou em todas as minhas decisões. Ao meu pai por não poupar esforços para me ajudar, seja dirigindo para me levar ao local de coleta de dados, me auxiliando com a preparação da câmera fotográfica ou demonstrando o mesmo entusiasmo que eu toda vez que eu compartilhava meus avanços no projeto. À minha mãe e meus irmãos pelo incentivo para que eu me inscrevesse no programa de mestrado.

Gostaria de agradecer também ao meu orientador Rogério Atem por aceitar o convite para me orientar neste trabalho, pelos conselhos e sugestões.

Agradecimento em especial ao meu coorientador William Vianna, pela contribuição técnica, paciência e disponibilidade.

Também não poderia deixar de agradecer à toda equipe da pousada recanto do lago, que desde o primeiro contato se mostrou prontamente disposta a ajudar e autorizou a obtenção das imagens do cultivo de cogumelos da sua propriedade. Muito obrigada por toda atenção e acolhimento.

*“No que diz respeito ao desempenho,
ao compromisso, ao esforço, à dedicação,
não existe meio termo.
Ou você faz uma coisa bem feita
ou não faz.”
(Ayrton Senna)*

RESUMO

A necessidade constante de uma maior produtividade e ao mesmo tempo maior qualidade nos produtos na área da agroindústria, exige que produtores agrícolas utilizem cada vez mais equipamentos tecnológicos para monitorar suas produções. Entre as ferramentas disponíveis para auxiliar a suprir esta demanda por métodos eficientes e seguros de produção de alimentos se destaca a visão computacional. Apesar do número de pesquisas sobre estas tecnologias para aumentar a produtividade e melhorar a qualidade dos produtos agrícolas crescer a cada dia, poucas delas investigam o uso da visão computacional especificamente na fungicultura. Com este propósito, um sistema de visão computacional para estimação de taxa de crescimento e previsão de colheita de cogumelos é proposto neste trabalho. O algoritmo foi desenvolvido utilizando a tecnologia python OpenCV. Primeiramente foram obtidas as imagens das camas de crescimento que passaram por um pré-processamento para refinamento da qualidade, em seguida foram efetuadas a diferenciação das regiões de interesse e técnicas de pós-processamento. Por fim foi realizada a extração das informações. Também foi desenvolvida uma interface de comunicação para transmitir estas informações ao agricultor utilizando a ferramenta Node-RED. As análises dos resultados mostram que o sistema proposto tem um bom desempenho em termos de identificação e medição de diâmetro de cogumelos. Foi alcançada uma precisão média de 86,4% na identificação dos cogumelos e 74,2% na determinação dos diâmetros dos mesmos.

Palavras-chave: Visão computacional. Cultivo de cogumelos. Python OpenCV. NodeRED.

ABSTRACT

The constant need for greater productivity and at the same time higher quality of products in the agribusiness area, demands that agricultural producers increasingly use technological equipment to monitor their productions. Among the tools available to help meet this demand for efficient and safe food production methods is the computer vision. Although research on these technologies to increase productivity and improve the quality of agricultural products is growing every day, few of them investigate the use of computer vision specifically in fungiculture. For this purpose, a computer vision system for growth rate estimation and mushroom harvest prediction is proposed in this work. The algorithm was developed using OpenCV python technology. First, images of the growth beds were obtained. These images underwent pre-processing for quality refinement. Subsequently, the regions of interest were differentiated and post-processing techniques were implemented. Finally, information extraction was performed. A communication interface was also developed to transmit this information to the farmer using the Node-RED tool. Analysis of the results shows that the proposed system performs well in terms of mushroom identification and diameter measurement. An average precision of 86.4 % was achieved in mushroom identification and 74.2 % in mushroom diameter determination.

Keywords: Computer vision. Mushroom cultivation. Python OpenCV. NodeRED.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de Venn	18
Figura 2 – Etapas do pré-processamento (da esquerda para direita: filtro bilateral, filtro convolucional, conversão para escalas de cinza)	31
Figura 3 – Resultado da etapa de segmentação	32
Figura 4 – Resultado das etapas de erosão, dilatação e fechamento (da esquerda para direita)	32
Figura 5 – Resultado da transformada distância	33
Figura 6 – Resultado do thresholding aplicado para separar os cogumelos agrupados . .	34
Figura 7 – Resultado final da identificação dos cogumelos	34
Figura 8 – Nó inject	35
Figura 9 – Nó media	36
Figura 10 – Nó exec	36
Figura 11 – Nó json	36
Figura 12 – Nó function	37
Figura 13 – Nó debug	37
Figura 14 – Nó text	37
Figura 15 – Nó gauge	38
Figura 16 – Nó LED	38
Figura 17 – Nó <i>chart</i>	38
Figura 18 – Resultado da identificação dos cogumelos	40
Figura 19 – Fluxo criado no Node-Red para gerar a IHM	41
Figura 20 – IHM desenvolvida com a ferramenta Node-Red	42
Figura 21 – Quadro da chave de busca utilizada na base Scopus	50
Figura 22 – Diagrama de Venn	52
Figura 23 – Distribuição da produção científica durante os anos	53
Figura 24 – Distribuição do número de publicações com os temas em questão ao longo dos anos	54
Figura 25 – Número de publicações por autor	55
Figura 26 – Rede de Citação. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de documentos de um autor: 3. Número mínimo de citações: 20.	58
Figura 27 – Rede de Cocitação. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de citações: 20.	59

Figura 28 – Rede de coautoria. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de documentos para um autor: 1. Número mínimo de citações por autor: 0. 60

LISTA DE TABELAS

Tabela 1 – Precisão alcançada para cada cama de crescimento analisada	39
Tabela 2 – Número de artigos publicados por periódico	54
Tabela 3 – Autores com o maior número de publicações e seus artigos mais citados . .	56
Tabela 4 – Artigos mais citados	57

LISTA DE ABREVIATURAS E SIGLAS

AP	<i>Agricultura de Precisão</i>
ROI	<i>Region of Interest</i>
TH	<i>Transformada de Hough</i>
MATLAB	Matrix Laboratory
LabVIEW	Laboratory Virtual Instrument Engineering Workbench
CLFN	<i>Call Library Function Node</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
IP	<i>Internet Protocol</i>
YOLO	<i>You Only Look Once</i>
RNA	<i>Redes Neurais Artificiais</i>
IHM	<i>Interface Homem-Máquina</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivos	17
1.1.1	Objetivo Geral	17
1.1.2	Objetivos Específicos	17
1.2	Justificativa	18
1.3	Organização da Dissertação	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	Processamento Digital de Imagens	20
2.1.1	Segmentação de Imagens	20
2.1.2	Tecnologias Utilizadas no Processamento Digital de Imagens	24
2.2	Visão Computacional	26
2.2.1	Visão Computacional na Agricultura de Precisão	26
3	TRABALHOS RELACIONADOS	27
3.1	Localização e Identificação de Frutas e Vegetais	27
3.2	Inspeção e Classificação de Cogumelos	28
3.3	Segmentação Celular	29
4	METODOLOGIA	30
4.1	Aquisição de Imagens	30
4.2	Pré-processamento	30
4.3	Segmentação	31
4.4	Pós-processamento	32
4.5	Análise	34
4.6	Interface Homem - Máquina	35
5	RESULTADOS	39
6	CONCLUSÃO	43
	REFERÊNCIAS	44

APÊNDICES **49**

**APÊNDICE A – VISÃO COMPUTACIONAL NO CULTIVO DE COGUMELOS:
UMA ANÁLISE BIBLIOMÉTRICA DA BASE DE CONHECI-
MENTO** **50**

A.1 Procedimentos metodológicos **50**

A.2 Análise Bibliométrica **52**

A.2.1 Levantamento da cronologia da produção 52

A.2.2 Identificação dos periódicos de destaque no tema 53

A.2.3 Identificação dos autores com maior número de publicações 55

A.2.4 Artigos com maior número de citações 56

A.2.5 Análises de citação, cocitação e coautoria 58

A.3 Considerações finais **61**

APÊNDICE B – CÓDIGO **62**

1 INTRODUÇÃO

A agricultura sustentável e precisa é a última tendência da sociedade contemporânea. Esta fornece bens agrícolas com o mínimo impacto ambiental e as mais altas taxas de produtividade possíveis (AIKATERINI, 2016). Atualmente, a maioria das terras adequadas para agricultura já estão em uso e grande parte da população está se preocupando mais com a forma como esses produtos estão sendo produzidos, por isso, para aumentar o volume de produção com os recursos limitados de cultivo, é preciso melhorar a eficiência através da agricultura de precisão (CHEN et al., 2019; RADCLIFFE; COX; BULANON, 2018).

“As pessoas, assim como os governos, estão se preocupando cada vez mais com os produtos químicos usados em produtos alimentícios, bem como com os métodos implementados para semear e colher. Além disso, o público e os agricultores estão começando a se preocupar com os impactos ambientais da agricultura, como o aprimoramento sintético das culturas, o controle severo das ervas daninhas e a rega de precisão (CHEN et al., 2019; RADCLIFFE; COX; BULANON, 2018).”

“A Agricultura de Precisão (AP) atualmente é um conceito muito importante para o desenvolvimento e expansão do agronegócio a nível mundial. A necessidade constante de uma maior produtividade e conseqüentemente na obtenção de qualidade nos produtos na área da agroindústria, exige que produtores agrícolas utilizem cada vez mais equipamentos tecnológicos para monitorização das suas produções, que permitem obter informações do estado atual do seu desenvolvimento e identificar possíveis problemas das mesmas. [...] Deste modo tornam-se mais competitivos no mercado, enquanto que aqueles que não usam a tecnologia a seu favor acabam por perder capacidade tornando-se menos competitivos (COSTA et al., 2018).”

Entre as ferramentas disponíveis para auxiliar a suprir esta demanda por métodos eficientes e seguros de produção de alimentos se destaca a visão computacional. O uso deste tipo de solução em setores agrícolas vem crescendo nos últimos anos devido à sua capacidade de reconhecer tamanho, forma, cor e textura de objetos de forma precisa, consistente e não destrutiva. Suas aplicações são diversas e incluem áreas como inspeção e triagem de frutas e vegetais, sistemas de locomoção de veículos agrícolas, colheita automatizada, aplicação localizada de herbicidas, gerenciamento de ervas daninhas, entre outras.

Do ponto de vista dos agricultores, a automação dessas atividades é extremamente benéfica, já que economizam tempo e dinheiro. Tradicionalmente elas são realizadas por mão-de-obra humana especializada. No entanto, na maioria dos casos, consomem muito tempo e exigem

muito trabalho. Além disso, quando se diz respeito às atividades com certo grau de subjetividade como a inspeção de qualidade, triagem de frutas e verduras por tamanho e definição de tempo certo de colheita, a visão computacional se apresenta como uma alternativa mais precisa e objetiva.

Segundo Marota et al. (2017):

“O uso de plataformas de gravação de dados ou fenotipagem totalmente automatizadas com base em imagens digitais, sensores, sons, sistemas não tripulados e visão computacional não invasiva em tempo real estão ganhando força e têm grande potencial para melhorar a qualidade do produto, práticas de gerenciamento, bem estar, desenvolvimento sustentável e saúde animal, contribuindo para melhorar a saúde humana.”

No setor de produção de cogumelos as aplicações de técnicas de visão de máquina são empregadas especialmente na identificação de aparência, como tamanho, forma e cor. Como os cogumelos maiores e sem imperfeições são vendidos por um preço melhor, muitos agricultores tem investido em sistemas de processamento de imagem para monitorização e inspeção durante o período de produção e colheita (LU et al., 2019). Através dos dados obtidos pode-se calcular a taxa de crescimento, estatísticas de quantidade, classificação de tamanho e qualidade, permitindo detectar problemas antecipadamente e com isto, melhorar a qualidade do produto. Estes dados ainda podem ser usados como base para definir parâmetros para o sistema de controle ambiental de estufas. Isto ajuda significativamente no aumento da eficiência, velocidade e rendimento da produção.

O consumo de cogumelos no Brasil se expandiu com o crescimento das colônias orientais e hoje é um alimento muito utilizado no preparo de pratos do cotidiano, especialmente daqueles mais sofisticados (FILHO; CAVALINI, 2015). O crescimento do vegetarianismo e do veganismo também contribuem para o aumento do seu consumo a nível mundial. Por apresentar grande quantidades de proteínas e aminoácidos eles são frequentemente utilizados como substitutos da carne.

Devido a esta expansão, tecnologias de processamento de imagem tem sido amplamente empregadas em aplicações relacionadas ao seu cultivo, como em: Development of a Mushroom Growth Measurement System Applying Deep Learning for Image Recognition (LU et al., 2019), Intelligent Mushroom Harvest Prediction System Proposal (COSTA et al., 2018) e An automatic sorting system for fresh white button mushrooms based on image processing (WANG et al., 2018).

Apesar da tecnologia de visão computacional aplicada ao monitoramento e inspeção durante o período de produção e colheita se mostrar relativamente madura, existem poucos estudos sobre sua aplicação na previsão de colheita de cogumelos. O presente estudo visa desenvolver um sistema de visão computacional para medição de crescimento e previsão de

colheita que deverá identificar os cogumelos, estimar a quantidade presente, seus diâmetros, calcular a taxa de crescimento e utilizando estas informações, estimar o tempo certo de colheita. Também será desenvolvida uma interface de comunicação para transmitir estas informações ao agricultor de forma mais visual e mais fácil de ser compreendida. Os dados de cada lote de produção poderão ser utilizados como base para o ajuste de variáveis de temperatura e umidade de um sistema de controle.

1.1 Objetivos

1.1.1 Objetivo Geral

O principal objetivo deste trabalho é desenvolver um sistema com visão computacional capaz de estimar o crescimento e determinar a previsão de colheita de cogumelos tipo paris (*Agaricus bisporus*).

1.1.2 Objetivos Específicos

Para alcançar o que foi proposto no objetivo geral foram definidos os seguintes objetivos específicos:

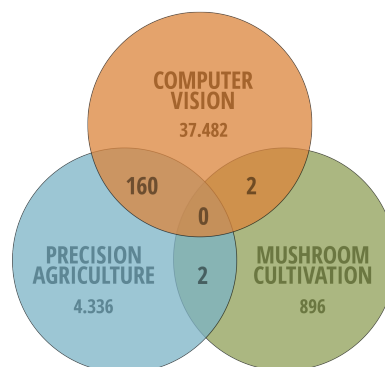
- a) Estudar as técnicas de processamento de imagem que possam auxiliar nas medições dos cogumelos nas camas de crescimento;
- b) Determinar a tecnologia que será utilizada para o processamento das imagens;
- c) Desenvolver o sistema com visão computacional com código embarcado;
- d) Realizar testes em ambiente controlado para estatisticamente determinar tamanho, taxa de crescimento e previsão de colheita em função de parâmetros de entrada;
- e) Desenvolver interface para utilização do sistema com visão computacional.

1.2 Justificativa

O cultivo comercial de cogumelos experimentou um crescimento explosivo nos últimos anos (PANAYI; PETERS; KYRIAKIDES, 2017). Somente nos Estados Unidos, o volume de vendas da colheita no período de 2018-2019 totalizou 846 milhões de libras (MUSHROOMS, 2019). No Brasil, a produção de cogumelos está em fase de expansão, sendo o consumo “per capita” no País de 288g/ano, com perspectivas de aumento (SANTANA, 2019). A China é o maior produtor mundial de cogumelos comestíveis, alcançando um rendimento anual estimado de 38,42 milhões de toneladas (FANG et al., 2018). Dados como estes comprovam que o setor encontra-se em plena expansão. Porém, apesar do número de pesquisas sobre tecnologias de visão computacional para aumentar a produtividade e melhorar a qualidade dos produtos agrícolas crescer a cada dia, poucas delas investigam o uso da visão computacional especificamente na fungicultura.

Na análise bibliométrica realizada a fim de selecionar os artigos a serem utilizados como base para este estudo foi possível constatar que esta é uma relação ainda pouco explorada na pesquisa científica. A pesquisa na amostra indexada na base de dados Scopus relacionando simultaneamente os conceitos “Computer Vision”, “Precision Agriculture” e “Mushroom Cultivation” não retornou nenhum trabalho e quando relacionando apenas os conceitos “Computer Vision” e “Mushroom Cultivation” retornou apenas dois estudos, como pode ser visto no diagrama de Venn representado na Figura 1.

Figura 1 – Diagrama de Venn



Fonte: Autor.

Uma das características fundamentais na avaliação da qualidade do produto é o seu diâmetro. Cogumelos com diâmetro acima de um determinado tamanho são vendidos por um melhor preço. Além disso, no gerenciamento da produção, os agricultores também devem considerar possíveis problemas, como fermentação do composto, corrida de semente e corpos de

frutificação, no processo de produção com base nos resultados de crescimento dos cogumelos (LU et al., 2019). Por isso, acabam utilizando muita mão-de-obra para monitorar o tamanho dos cogumelos durante a formação do corpo frutífero e encontrar aqueles com o tamanho ideal para colheita. Esses resultados podem ser imprecisos quando medidos e estimados manualmente. Os cogumelos crescem em taxas e intervalos irregulares, desta forma a colheita envolve uma série de tarefas consecutivas. Assim, existe um mercado claro para o desenvolvimento de sistemas precisos capazes de localizar e identificar cogumelos, acompanhar a sua taxa de crescimento e prever o tempo certo de colheita.

1.3 Organização da Dissertação

O estudo aqui apresentado foi sistematizado de forma a orientar o conhecimento acerca da temática abordada. O conteúdo de cada capítulo é detalhado a seguir:

- a) **Capítulo 1** – Apresenta uma contextualização do tema abordado, os objetivos e justificativa do trabalho;
- b) **Capítulo 2** – Trata da fundamentação teórica através da formalização de técnicas de visão computacional e processamento digital de imagens;
- c) **Capítulo 3** – Aborda as produções científica relacionados ao método proposto;
- d) **Capítulo 4** – Descreve os procedimentos metodológicos utilizados para a realização do trabalho;
- e) **Capítulo 5** – Compreende as análises dos resultados obtidos;
- f) **Capítulo 6** – Apresenta as conclusões e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Processamento Digital de Imagens

O processamento digital de imagens pode ser definido como a transformação de dados de uma imagem analógica em dados digitais (ou seja, 0 e 1) usando computadores para analisá-los e entendê-los (CHOUHAN; SINGH; JAIN, 2019). É a etapa mais importante na utilização da visão computacional, já que ela permite a retirada da informação relevante para as futuras análises através da aplicação de algoritmos específicos em função de características como cor, textura, forma, geometria ou padrão.

“Os dados e informações na agricultura se originam principalmente de imagens fotográficas, mas matematicamente para estimar ou processar dados fotográficos, é um desafio. Portanto, a tecnologia de processamento de imagem digital ajuda a processar imagens e tenta uma extensão para sua análise. [...] Como a ciência da informação está crescendo rapidamente, o reconhecimento de padrões baseado em visão computacional e o processamento de imagens são técnicas amadurecidas para análise de segurança e qualidade de várias aplicações agrícolas (BHARGAVA; BANSAL, 2018).”

Técnicas de processamento de imagem podem ser usadas na classificação, controle de qualidade e determinação do tamanho de um cogumelo (SERT; OKUMUS, 2014). Primeiramente a imagem é pré-processada. Esta etapa não se destina a extrair informações da imagem, mas a remover falhas como ruídos, distorções geométricas, falta de nitidez e iluminação não adequada. A seguir é efetuada a segmentação que é normalmente usada para localizar objetos e contornos nas imagens (PATEL et al., 2012). É a etapa mais complexa e desafiadora do processamento de imagens digitais e fica ainda mais complicada quando a aquisição das imagens ocorre em condições não padronizadas com diferentes característica de iluminação, ângulo de captura, foco e etc. Após a segmentação é realizado o pós-processamento que é responsável por corrigir possíveis defeitos ainda presentes na imagem segmentada. Por fim, são extraídas as informações de interesse.

2.1.1 Segmentação de Imagens

A segmentação é uma técnica utilizada no processamento digital de imagens com o objetivo de dividir a imagem em partes que possuem características e propriedades similares para representá-la de maneira mais significativa e mais fácil de analisar.

Quando se trata de segmentação não existe uma abordagem universal que suporte todos os tipos de imagens, pois esse método subjetivo falha em detectar incertezas. Os métodos de segmentação dependem principalmente da aplicação (MAHATA et al., 2017). Levando em consideração as premissas do problema em questão relacionadas à forma, localização e distribuição de intensidade é possível partir de um conjunto específico de técnicas que são mais prováveis de serem úteis na solução do problema.

Devido à similaridades de características como apresentação de formato circular, presença de aglomeração e sobreposição de elementos, diversas das técnicas empregadas na análise de imagens médicas, mais especificamente na área de segmentação celular, são também utilizadas na detecção de cogumelos.

Algumas das técnicas de segmentação de imagens são descritas a seguir:

A - *Thresholding*

Os métodos de *thresholding* são os mais simples, porém eficazes para segmentação de imagens através da conversão de imagens em escalas de cinza em binárias. São muito aplicáveis quando o objeto de interesse deve ser separado do plano de fundo e na presença de altos níveis de contraste. A segmentação é realizada usando um valor limite, definido com base no valor da intensidade que separa a região de interesse (ROI) do plano de fundo (GILL; GIRDHAR; SINGH, 2019). Existem três tipos de *thresholding*:

1. *thresholding* global - Utiliza um único valor limiar (T) para a imagem inteira, baseando-se no pressuposto de que a imagem apresenta um histograma bimodal. Se o valor do pixel for maior que o valor limiar, é atribuído um valor, caso contrário, é atribuído outro.
2. *thresholding* variável - Nesse tipo de *thresholding*, o valor de T pode variar. T assume valores diferentes para diferentes regiões da mesma imagem, o que proporciona um melhor resultado na presença de diferentes condições de iluminação.
3. *thresholding* múltiplo - O valor do limiar (T) é fixo, porém existe mais de um limiar (T1, T2, T3 ...)

Apesar das abordagens baseadas em *thresholding* serem amplamente utilizadas, elas também podem apresentar desvantagens como a perda de informação após o processo de binarização e a falta de especificidade para uma classificação precisa.

B - Operações morfológicas

O processamento morfológico de imagens consiste em modificá-las com base em estruturas geométricas. A filtragem é realizada examinando as estruturas geométricas e topológicas de objetos com uma forma predefinida (XING; YANG, 2016). Um modelo chamado elemento estruturador é situado em todos os locais possíveis na imagem e comparado com a região subsequente de pixels. Algumas operações testam se o elemento “se encaixa” na região, enquanto

outras testam se ele “atinge” ou cruza a região (SHARMA; SHARMA, 2013). Estes tipos de operações se concentram mais em imagens binárias, mas também podem ser aplicadas a imagens em escala de cinza.

A base da operação morfológica são as operações de dilatação, erosão, abertura e fechamento.

Dilatação - A dilatação adiciona pixels às bordas dos objetos provocando um aumento de tamanho dos mesmos e fazendo com que buracos nesta região se tornem menores. Esta operação toma duas partes como dados. A primeira é a imagem de entrada a ser dilatada e a segunda é o elemento estruturador também conhecido como kernel que determina o quanto a imagem deve ser dilatada (CHUDASAMA et al., 2015).

Erosão - A erosão remove pixels das bordas dos objetos provocando uma diminuição no tamanho dos mesmos e fazendo com que buracos nesta região se tornem maiores. A erosão, assim como a dilatação toma duas partes como dados. A primeira é a imagem de entrada a ser corroída e a segunda é o kernel que determina o quanto a imagem deve ser erodida (CHUDASAMA et al., 2015).

Abertura - A abertura consiste em uma erosão seguida de uma dilatação e é usada para erradicar todos os pixels em regiões muito pequenas para incluir o elemento estruturador (SHARMA; SHARMA, 2013).

Fechamento - O fechamento compreende uma dilatação seguida de erosão e pode ser usado para preencher buracos e pequenos espaços (SHARMA; SHARMA, 2013).

C - Transformada de Watershed

O algoritmo de segmentação de watershed é baseado na representação de uma imagem na forma de um relevo topográfico, em que o valor de cada elemento da imagem caracteriza sua altura neste ponto (KORNILOV; SAFONOV, 2018).

Segundo Bieniek e Moga (2000):

“O algoritmo tradicional simula um processo de inundação. Assim, uma imagem é identificada com uma superfície topográfica, na qual a altitude de cada ponto é igual ao nível de cinza do pixel correspondente. Buracos são então perfurados em todos os mínimos regionais do relevo (platôs conectados de altitude constante a partir dos quais é impossível alcançar um local de altitude mais baixa sem ter que subir). Afundando lentamente toda a superfície em um lago, a água brota pelos buracos e mergulha progressivamente nas paredes adjacentes. Para impedir que fluxos de água provenientes de diferentes orifícios se misturem, um obstáculo é configurado nos locais da reunião. Quando o relevo é completamente coberto pela água, o conjunto de obstáculos mostra a imagem de watershed.”

Atualmente, esta técnica é usada com grande frequência para a segmentação de imagens por ser um algoritmo relativamente rápido e permitir uma segmentação precisa retendo recursos como a definição dos contornos nas imagens. No entanto, a segmentação baseada na transformada de watershed é sensível a ruídos e muitas vezes pode apresentar segmentação excessiva.

D - Transformada de Hough

A transformada de Hough (HOUGH, 1962) foi patenteada por Paul V. C. Hough em 1962 como uma técnica matemática para a localização de retas. Posteriormente, o método foi estendido para que pudesse ser utilizado para curvas mais genéricas como círculos e elipses.

“A idéia por trás do método é simples: formas paramétricas em uma imagem são detectadas procurando pontos de acumulação no espaço de parâmetros. Se uma forma específica estiver presente na imagem, o mapeamento de todos os seus pontos no espaço de parâmetros deverá se agrupar em torno dos valores dos parâmetros que correspondem a essa forma (ANTOLOVIC, 2008).”

A transformada de Hough (TH) clássica pode ser aplicada a muitos problemas de visão computacional, pois a maioria das imagens contém bordas que podem ser descritas por curvas regulares (CHANDRASEKAR; DURGA, 2014). Sua principal vantagem é a tolerância a falhas na continuidade dos limites dos contornos dos objetos a serem detectados, no entanto, a TH pode gerar picos falsos devido a ruídos na imagem, extração incorreta de bordas ou presença de objetos que se tocam (XING; YANG, 2016).

E - Agrupamento (*Clustering*)

A idéia central por trás do método de segmentação baseada em *clustering* é particionar os pixels da imagem em vários *clusters*, dependendo da variação nas propriedades do pixel, visando categorizar os pixels em tipos semelhantes (GILL; GIRDHAR; SINGH, 2019). Segundo Kaur e Kaur (2014) existem duas categorias gerais de métodos de agrupamento:

1. *hard clustering* - divide a imagem em um conjunto de *clusters*, de forma que um pixel só possa pertencer a um *cluster*. Esses métodos usam funções de associação com valores 1 ou 0, ou seja, um determinado pixel pode pertencer a um *cluster* específico ou não.
2. *soft clustering* - é o tipo mais natural de agrupamento porque na vida real a divisão exata não é possível devido à presença de ruído. Os pixels são particionados em *clusters* com base na associação parcial, isto é, um pixel pode pertencer a mais de um *cluster*.

A segmentação baseada em *cluster* fornece resultados precisos, mas as principais desvantagens são que elas são computacionalmente lentas e determinar os limites não é uma tarefa trivial (GILL; GIRDHAR; SINGH, 2019).

Na segmentação de células ou cogumelos, a análise de agrupamento pode não gerar a segmentação final, mas fornece suporte para a extração subsequente das fronteiras dos objetos (XING; YANG, 2016).

A seleção de uma técnica de segmentação de imagens e a interpretação dos resultados obtidos é uma tarefa desafiadora. É necessário um conhecimento considerável e de domínio bem estabelecido. Na maioria das aplicações, para alcançar um desempenho satisfatório é necessário combinar algumas destas técnicas e as vezes até incluir outros métodos.

2.1.2 Tecnologias Utilizadas no Processamento Digital de Imagens

A - Matlab

O Matrix Laboratory (MATLAB) é um pacote de software interativo de alto desempenho desenvolvido pela MathWorks Inc. que permite operações matriciais, implementação de algoritmos, simulação, processamento de imagens e sinais e outras funções.

A Image Processing Toolbox TM fornece um conjunto abrangente de algoritmos e aplicativos de fluxo de trabalho com padrão de referência para processamento de imagens, análise, visualização e desenvolvimento de algoritmos (The MathWorks, Inc., 1994). Oferece a possibilidade de restauração de imagem ruidosa ou degradada e melhora a clareza da imagem, extração de características e análise de formas e texturas (GAO, 2013). Os aplicativos da Toolbox permitem automatizar fluxos de trabalho comuns de processamento de imagens. É possível segmentar interativamente dados de imagem, comparar técnicas de registro e processar grandes conjuntos de dados em lote (The MathWorks, Inc., 1994).

O ambiente Matlab é relativamente simples e amigável de usar, fornece vários tipos de funções e algoritmos e não é necessário se preocupar com alocação e vazamento de memória (GAO, 2013). Por outro lado, não é a tecnologia mais rápida e assim como o próprio software, a toolbox para processamento de imagens também precisa ser comprada.

B - Labview

Laboratory Virtual Instrument Engineering Workbench (LabVIEW) é uma plataforma de design de sistema e ambiente de desenvolvimento para uma linguagem de programação visual da National Instruments (MAHMOODI; JAMES; JOHANSEN, 2018).

O LabVIEW possui uma biblioteca destinada especificamente ao processamento de imagens. A IMAQ vision toolbox apresenta um conjunto completo de funções de processamento e aquisição de imagens digitais que melhoram a eficiência dos projetos e reduzem o esforço de programação dos usuários, obtendo melhores resultados em menor tempo (POSADA-GOMEZ et al., 2011).

A versatilidade fornecida pelo software e também a capacidade da toolbox IMAQ aumentam a probabilidade de utilização do processamento digital de imagens em qualquer tipo

de aplicação (NELLUTLA, 2018).

Segundo Ding, Zhang e Kan (2015),

“As principais vantagens do LabVIEW são a interface gráfica do usuário e a linguagem de programação gráfica (linguagem G) fácil de usar, que permite aos programadores implementar programas arrastando e soltando ícones rapidamente. O LabVIEW IMAQ Vision é uma tecnologia de inspeção relativamente nova, rápida e eficiente. No entanto, o LabVIEW não pode fornecer completamente os drivers necessários para os usuários e também não é bom no processamento de muitos dados. Porém Call Library Function Node (CLFN) pode ser usado para chamar arquivos de biblioteca de vínculo dinâmico para resolver os problemas acima.”

C - OpenCV

OpenCV é uma biblioteca ampla de visão computacional e aprendizado de máquina utilizada em uma grande variedade de contextos. Seu poder está relacionado à enorme quantidade de algoritmos de visão computacional clássicos e avançados fornecidos por ela (DOMÍNGUEZ; HERAS; PASCUAL, 2017; KAEHLER; BRADSKI, 2015). Suas áreas de aplicação incluem sistema de reconhecimento facial, reconhecimento de gestos, compreensão de movimento, detecção de objetos, rastreamento de movimento, aprendizado de máquina e análise de vídeo. Esta pode ser implementada com ligações em C ++, Python, Java e MATLAB / OCTAVE e wrappers em C , Pearl, Ch, Haskell e Ruby. Essas linguagens de programação podem ser usadas e desenvolvidas em qualquer sistema operacional (S; ARANGAIAH; THANGARAJ, 2018).

OpenCV-Python é uma biblioteca de ligações Python projetada para resolver problemas de visão computacional (K.; MORDVINTSEV, 2013). Esta foi escolhida para ser usada no presente trabalho por ser uma biblioteca de uso aberto, conter várias funções para processamento de imagens especializadas no tratamento de imagens em tempo real e usar interface para a linguagens de programação Python, que é uma linguagem simples que permite expressar as ideias em poucas linhas de código.

2.2 Visão Computacional

A visão de máquina compreende o uso de dispositivos para detecção óptica sem contato para receber e interpretar automaticamente a imagem de uma cena real, a fim de obter informações e / ou controlar máquinas ou processar imagens (PATEL et al., 2012; ZUECH, 2000). Normalmente este método apresenta a capacidade de fazer o reconhecimento em imagens através da aplicação de técnicas e algoritmos que permitem extrair e apresentar a informação relevante e pretendida (COSTA et al., 2018). Esta tecnologia não apenas fornece um alto nível de flexibilidade e repetibilidade a um custo relativamente baixo, mas também, permite uma produtividade bastante alta sem comprometer a precisão (PATEL et al., 2012).

O conceito subjacente tem vastas áreas de aplicações em sensoriamento remoto, processamento de imagens médicas, agricultura inteligente ou de precisão, imagem de satélite, defesa, etc (CHOUHAN; SINGH; JAIN, 2019). O método inclui o modo como é feita a captura da imagem, o processamento e análise das mesmas (COSTA et al., 2018).

2.2.1 Visão Computacional na Agricultura de Precisão

Na agricultura, o sistema de visão computacional e o processamento de imagens estão crescendo prontamente na área de pesquisa, o que é uma técnica de análise significativa para pré e pós-colheita das culturas (BHARGAVA; BANSAL, 2018).

“Dispositivos inteligentes, como câmeras e sensores, são colocados nos campos para coletar as informações que são transmitida ao sistema de suporte à decisão com a ajuda da internet. Estas informações coletadas são então analisadas para a tomada de decisão em gestão e operações para melhoria da produtividade e qualidade da colheita.” (CHOUHAN; SINGH; JAIN, 2019).

A maioria dos estudos na área são focados na inspeção de qualidade e classificação do produto. Porém a aplicação de sistemas de visão computacional também é possível em outras áreas da agricultura, como na orientação de máquinas agrícolas, identificação de doenças em frutas e vegetais, análise de crescimento, identificação de ervas daninhas e aplicação localizada de fertilizantes.

3 TRABALHOS RELACIONADOS

Nos últimos anos, foram realizados diversos estudos na área de inspeção e classificação de produtos agrícolas e alimentares por sistemas de visão computacional, porém estudos voltados para a fungicultura são escassos. Por isso, nesta seção serão discutidos não só trabalhos na área de sistemas de inspeção e classificação de cogumelos, mas também estudos nas áreas de localização e identificação de frutas e vegetais e também estudos na área de segmentação celular que utilizam métodos semelhantes aos usados para segmentar e reconhecer cogumelos.

3.1 Localização e Identificação de Frutas e Vegetais

A visão computacional tem sido amplamente utilizada para a inspeção e classificação de frutas e vegetais. Ela oferece o potencial de automatizar práticas de classificação manual e, assim, padronizar técnicas e eliminar tarefas de inspeção tediosas.

Uma revisão da literatura sobre a avaliação da qualidade de frutas e vegetais utilizando visão computacional foi desenvolvida por Bhargava e Bansal (2018). Os autores apresentam uma visão geral de vários métodos como por exemplo pré-processamento, segmentação, extração de características e classificação que abordam qualidade de frutas e vegetais com base em cor, textura, tamanho, forma e defeitos. Também comparam algoritmos para inspeção de qualidade propostos por diferentes autores e sugerem formas de melhorar a performance dos mesmos. Segundo eles, o desempenho dos sistemas podem melhorar considerando as imagens de frutas e vegetais capturadas de diferentes direções. Ainda seria possível explorar a combinação de espaços de cores e incluir imagens de diferentes regiões. Um sistema generalizado também pode ser projetado para classificar e detectar defeitos em várias frutas e vegetais ao invés da verificação de uma única fruta.

Outro estudo para contar a quantidade de frutas verdes em laranjeiras foi proposto por Maldonado e Barbosa (2016) que com base na correlação conhecida entre o número de frutos visíveis em uma imagem digital e o total de frutos presentes em uma laranjeira, desenvolveram um método para a extração das características dos frutos verdes com uma combinação das técnicas de conversão do modelo de cores, limiar, equalização do histograma, filtragem espacial com operadores Laplace e Sobel e desfoque gaussiano. Além disso, construíram e testaram um algoritmo para reconhecê-los e contá-los, com taxas de detecção de falsos positivos de 3% em imagens adquiridas em boas condições. Com o algoritmo é possível estimar o número médio de frutos visíveis nas árvores dentro de um erro tolerado de 5%, com até 46 imagens em aproximadamente 8 minutos sem interação humana.

3.2 Inspeção e Classificação de Cogumelos

A classificação manual de cogumelos apresenta muitas desvantagens, como grande quantidade de trabalho, baixa produtividade, diferentes padrões de classificação pessoal e precisão de classificação instável (WANG et al., 2019). Por outro lado, os sistemas baseados em visão computacional são uma das maneiras eficazes de examinar recursos visuais e detectar o tamanho de um cogumelo.

No artigo intitulado “*Development of a Mushroom Growth Measurement System Applying Deep Learning for Image Recognition*” (LU et al., 2019), os autores desenvolvem um sistema para medir o tamanho e contar a quantidade de cogumelos. De acordo com os resultados de cada medição, a taxa de crescimento é estimada. O sistema proposto também registra os dados dos e os transmite ao telefone celular do agricultor, o que melhora a eficácia do gerenciamento da produção. A linguagem C++ foi usada para construir o algoritmo, funções e interface homem-máquina. Os sistemas operacionais Microsoft Windows e MQTT (*Message Queuing Telemetry Transport*) Broker foram usados para transmitir as mensagens. As imagens da estufa foram capturadas por uma câmera de rede IP e os cogumelos foram reconhecidos e localizados usando o algoritmo YOLO v3. Um algoritmo de correção de posicionamento também foi proposto para melhorar a precisão da localização dos cogumelos. Os experimentos mostraram que o sistema apresenta um bom desempenho.

Já Wang et al. (2018) focam no processo de classificação pós colheita. Em “*An automatic sorting system for fresh white button mushrooms based on image processing*” eles descrevem um sistema de triagem automática para cogumelos frescos brancos com base no processamento de imagens de acordo com o diâmetro do chapéu dos cogumelos. Primeiro, é projetado um sistema de hardware de classificação automática que consiste em um mecanismo de transporte, um sistema de aquisição de imagens, um módulo de controle e um atuador. Em segundo lugar, apresentam um algoritmo de imagem baseado no método de *watershed*, operador Canny, operação OR e operação fechada para determinar o diâmetro do chapéu de cogumelos brancos frescos. Em seguida, uma estratégia de controle precisa é introduzida e por fim, apresentam o software de análise e controle baseado no OpenCV 2.4.10 e no Visual Studio 2010. Na parte experimental, um sistema de protótipo foi testado para validar sua aplicabilidade e confiabilidade pelo desempenho e resultados da classificação. Os resultados experimentais mostraram que o sistema obtém classificação e triagem automáticas on-line de cogumelos brancos frescos com destruição mínima.

O trabalho mais próximo ao proposto na presente pesquisa é apresentado por Costa et al. (2018) em “*Intelligent Mushroom Harvest Prediction System Proposal*”. Os autores propõem um Sistema Inteligente de Previsão de Colheita de Cogumelos (SIPCC), baseado em técnicas e métodos de visão por computador e Redes Neurais Artificiais (RNA). Na primeira fase, com base na análise funcional, foi proposta a arquitetura do sistema, SIPCC, de forma simples e compreensível e na segunda fase foi construída a RNA para obter a previsão de colheita. Os

dados obtidos foram muito satisfatórios, obtendo-se quantidades em Kg muito próximas de quantidades analisadas de dados históricos.

3.3 Segmentação Celular

Devido à semelhança nas formas circulares dos chapéus dos cogumelos e dos corpos celulares, grande parte do trabalho realizado na comunidade de imagens biomédicas na segmentação celular é relevante também na segmentação de cogumelos. Além do formato circular, ambos apresentam aglomeração e sobreposição.

Os autores Zheng e Ahmad (2008) apresentam uma revisão dos métodos de segmentação de imagens de células. Primeiramente, fornecem uma visão geral das técnicas de segmentação de imagens baseadas em histogramas, bordas, região e pixels. Posteriormente, resumem quatro questões principais da segmentação de imagens de células: a lacuna semântica entre a segmentação heurística baseada no conhecimento realizada por humanos e o procedimento algorítmico realizado por computadores; fisionomia variável causada por ruído de fundo, coroa celular irregular devido ao desfoque da imagem e, finalmente, adesão celular. Por fim, chegam à conclusão de que o uso de uma única técnica não parece ser capaz de lidar com todos os problemas. Introduzem então as técnicas híbridas. Também discutem a importância de ter sistemas capazes de aprender a segmentar imagens, juntamente com o potencial de aprendizado não supervisionado.

Outra relevante revisão foi feita em “*A Review on Cell Detection and Segmentation in Microscopic Images*”. Os autores Thomas e John (2017) analisam vários métodos automatizados para detecção e segmentação de células e fornecem um resumo de cada um deles. Segundo eles ainda não foi introduzido nenhum método que seja genérico e possa ser treinado para uma ampla gama de aplicações com alta taxa de precisão e sensibilidade. Por isso, é necessário dar mais atenção à criação de um software automatizado, que utilizará com eficiência a tecnologia atual e levará essa área a uma dimensão totalmente diferente.

4 METODOLOGIA

Um sistema de visão computacional para estimação de taxa de crescimento e previsão de colheita de cogumelos é proposto neste trabalho. O algoritmo foi desenvolvido utilizando a tecnologia python OpenCV e todo o sistema consiste basicamente de cinco etapas. Primeiramente foram obtidas as imagens das camas de crescimento que passaram por um pré-processamento para refinamento da qualidade, em seguida foram efetuadas a diferenciação das regiões de interesse e técnicas de pós-processamento. Por fim foi realizada a extração das informações. Também foi desenvolvida uma interface de comunicação para transmitir estas informações ao agricultor. Todo o processo é descrito de forma detalhada a seguir.

4.1 Aquisição de Imagens

Em aplicações alimentares, as ferramentas de aquisição de imagens utilizadas são câmera, ultrassom, ressonância magnética, tomografia elétrica e tomografia computadorizada (BHARGAVA; BANSAL, 2018). Neste projeto foi utilizada uma câmera fotográfica Canon modelo EOS Rebel T6 com configuração para retrato noturno. As fotos foram obtidas de um cultivo de cogumelos paris na cidade de Pedra Azul, ES. Quatro camas em diferentes localizações foram fotografadas com um intervalo de 48 horas e a uma distância de aproximadamente 41cm.

As imagens obtidas devem ser carregadas no programa desenvolvido baseado em Python, processadas e analisadas. Para que um programa python realize o processamento de imagens, a função OpenCV deve ser importada e o “cv2” deve ser anexado em cada código de linha (MEIMBAN et al., 2018).

4.2 Pré-processamento

A primeira etapa do pré-processamento consistiu na aplicação de um filtro bilateral à imagem com o objetivo de reduzir ruídos indesejados sem prejudicar a definição das bordas dos objetos. A função para este filtro em python utilizando OpenCV é mostrada a seguir:

$$cv2.bilateralFilter(src, d, sigmaColor, sigmaSpace[, dst[, borderType]]) \quad (4.1)$$

Para a identificação dos cogumelos, a sua aparência em relação ao plano de fundo desempenha um papel muito importante. Quanto mais significativa for a aparência, mais precisa será a identificação. Por isso, o segundo passo do algoritmo foi implementar um filtro convolucional para aumentar o contraste da imagem. Foi criada uma matriz (*kernel*) e aplicado o comando **cv2.filter2D**.

Em seguida, a imagem foi convertida para escalas de cinza para que pudesse ser usada na etapa seguinte de segmentação. O resultado destas operações é mostrado na [Figura 2](#).

Figura 2 – Etapas do pré-processamento (da esquerda para direita: filtro bilateral, filtro convolucional, conversão para escalas de cinza)



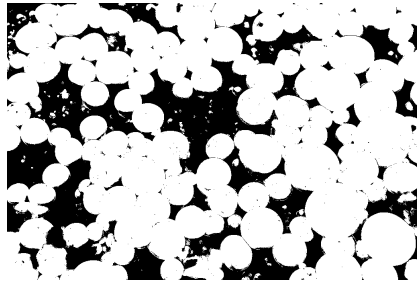
Fonte: Autor.

4.3 Segmentação

Uma vez que a imagem está livre de ruídos, os cogumelos precisam ser separados do plano de fundo. Isto pode ser feito através de técnicas de *thresholding*, que utilizam como critério o valor de intensidade dos pixels. Se o valor do pixel for maior que o valor de *thresholding*, é atribuído um valor (pode ser branco), caso contrário é atribuído outro (pode ser preto) ([MORDVINTSEV; K., 2013](#)).

Neste trabalho foi utilizado o *thresholding* de Otsu ([OTSU, 1979](#)). Este método calcula automaticamente o valor de *thresholding* a partir do histograma de uma imagem bimodal ([MORDVINTSEV; K., 2013](#)). O resultado do processo de segmentação é mostrado na [Figura 3](#).

Figura 3 – Resultado da etapa de segmentação



Fonte: Autor.

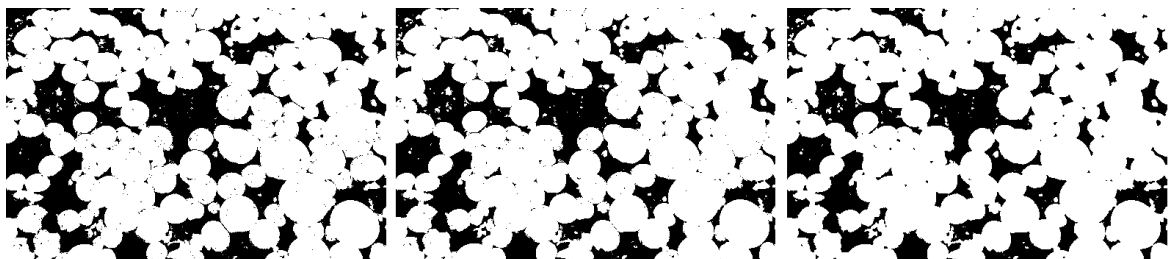
Embora seja possível identificar claramente a posição dos cogumelos na imagem obtida, ainda existem ruídos e buracos. Para superar esses problemas, uma série de métodos de pós-processamento são aplicados.

4.4 Pós-processamento

Para remover ruídos e buracos ainda presentes na imagem segmentada foram utilizadas as operações morfológicas erosão, dilatação e o fechamento. A erosão desempenha o papel de “encolher” e “afinar” os objetos na imagem, enquanto a dilatação costuma “crescer” e “engrossar” os objetos (TOMARI et al., 2015). Já o fechamento compreende uma dilatação seguida de erosão e pode ser usado para preencher buracos e pequenos espaços (SHARMA; SHARMA, 2013).

Neste projeto, uma sequência de uma iteração de erosão, uma iteração de dilatação e uma operação de fechamento com um *kernel* (3,3) foram implementadas. Os resultados são mostrados na Figura 4.

Figura 4 – Resultado das etapas de erosão, dilatação e fechamento (da esquerda para direita)



Fonte: Autor.

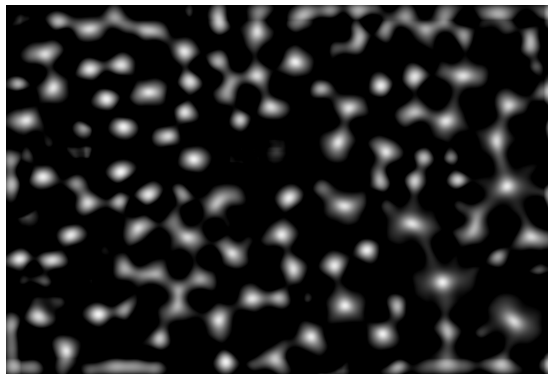
Em seguida, a função `cv2.getStructuringElement` foi utilizada para construir e retornar um elemento estruturador elíptico que foi usado em uma nova operação morfológica

de fechamento. Sobre o resultado destas operações aplica-se a transformada distância (FELZENSZWALB; HUTTENLOCHER, 2012). Segundo TOMARI et al. (2015).

A transformada distância é uma máscara operadora aplicada às imagens para determinar o mapa de distância dos objetos na imagem. O resultado da transformação é uma imagem em escala de cinza semelhante à imagem de entrada, exceto que as intensidades do nível de cinza dos pontos dentro das regiões do primeiro plano são alteradas para mostrar a distância até o limite mais próximo de cada ponto ao pixel zero mais próximo.

O resultado deste processo pode ser visto na [Figura 5](#).

Figura 5 – Resultado da transformada distância

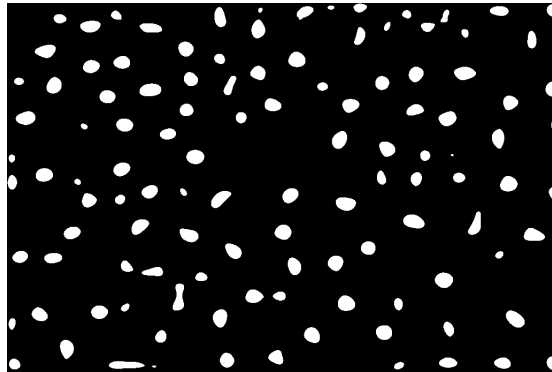


Fonte: Autor.

Pode-se observar que, a intensidade variou com o nível mais brilhante no centro do objeto e diminuiu lentamente à medida que atingia a borda do objeto.

Uma vez obtido o mapa de distância, a função `cv2.minMaxLoc` é utilizada para encontrar a máxima local da imagem resultante. A localização dos máximos corresponde aos centros dos círculos e os valores máximos correspondem aos seus raios. Um valor de *thresholding* é então aplicado para separar os cogumelos agrupados. O valor de *thresholding* selecionado corresponde ao valor do raio dos círculos encontrados com a função anterior multiplicado por um valor selecionado empiricamente. A [Figura 6](#) mostra a imagem obtida após a operação.

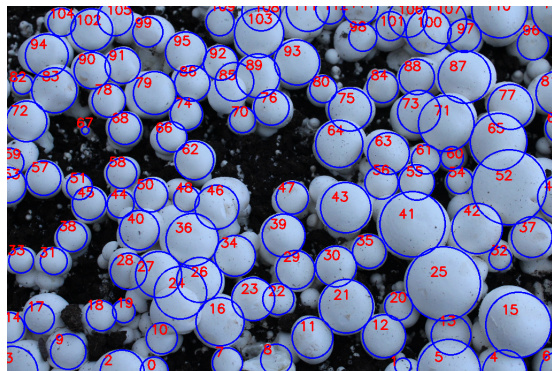
Figura 6 – Resultado do thresholding aplicado para separar os cogumelos agrupados



Fonte: Autor.

Por fim, as funções `cv2.findContours` e `cv2.circle` são aplicadas para encontrar e desenhar os círculos utilizados para localizar e estimar o diâmetro dos cogumelos. A Figura 7 mostra os cogumelos identificados após todos os processos.

Figura 7 – Resultado final da identificação dos cogumelos



Fonte: Autor.

4.5 Análise

Após a obtenção dos resultados de posicionamento do processamento da imagem, o número de cogumelos identificados pode ser calculado e seus diâmetros em pixels estimados. O valor de diâmetro em pixel (D_p) deverá então ser multiplicado pela altura do sensor da câmera e pela distância da câmera durante obtenção da imagem (D_c), e dividido pela distância focal (D_f) multiplicada pela altura da imagem em pixels (A_i) para se estimar o diâmetro real dos cogumelos (D_r) em mm.

$$Dr(mm) = \frac{Dp(pixels) * As(mm) * Dc(mm)}{Df(mm) * Ai(pixels)} \quad (4.2)$$

A taxa de crescimento, Tc (mm/h), é calculada pela mudança dimensional do cogumelo em um período de tempo e é indicada por:

$$Tc = \frac{(Sn - S1)}{I} \quad (4.3)$$

onde S1 é o primeiro resultado da medição, Sn é o enésimo resultado da medição e I é o período de tempo de S1 a Sn.

O tempo ideal de colheita é estimado através da porcentagem de cogumelos com diâmetro maior ou igual ao diâmetro ideal. Neste trabalho, o padrão adotado foi 50% dos cogumelos com diâmetro igual ou maior que 40mm.

4.6 Interface Homem - Máquina

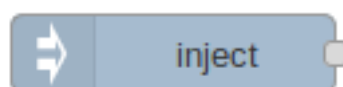
O código desenvolvido gera como saída um arquivo no formato json contendo: a quantidade de cogumelos identificados, o diâmetro médio, a quantidade de cogumelos em cada faixa de intervalo de diâmetro (0-15mm, 16-25mm e >25mm), a taxa de crescimento e a estimativa de colheita.

Estes dados são utilizados em uma interface de comunicação desenvolvida utilizando a ferramenta Node-RED para transmitir as informações ao produtor de forma mais visual e mais fácil de ser compreendida. Esta ferramenta faz uso de uma abordagem de programação visual que permite utilizar blocos de códigos predefinidos(conhecidos como nós) para executar determinadas tarefas.

Neste trabalho foram utilizados os seguintes nós:

1. *inject* - O nó *inject* pode ser usado para acionar manualmente um fluxo clicando no botão do nó no editor ou disparar fluxos automaticamente em intervalos regulares. Este nó foi utilizado no presente trabalho para inicializar a execução do código do programa a cada intervalo de 10 segundos e está representado na [Figura 8](#)

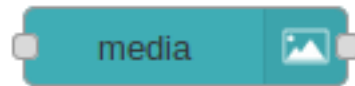
Figura 8 – Nó inject



Fonte: Autor.

2. *media* - Nó da interface do usuário Node-RED para mostrar arquivos de mídia (imagem e vídeo) no *dashboard*. Foi utilizado para mostrar as imagens mais recentes das camaras de crescimento monitoradas e está representado na [Figura 9](#)

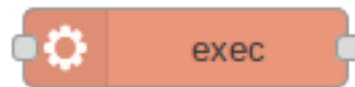
Figura 9 – Nó media



Fonte: Autor.

3. *exec* - Este nó executa um comando do sistema e retorna sua saída. Foi utilizado neste trabalho para executar o código desenvolvido para identificação dos cogumelos, cálculo da taxa de crescimento e previsão de colheita. Representado na [Figura 10](#)

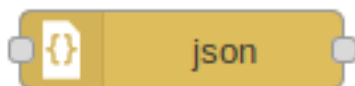
Figura 10 – Nó exec



Fonte: Autor.

4. *json* - Este nó é capaz de converter uma mensagem entre uma string JSON e um objeto JavaScript e vice-versa. Como a saída do código desenvolvido consiste de uma string json, este nó foi necessário para que a mensagem pudesse ser convertida em um objeto JavaScript e as variáveis pudessem ser acessadas individualmente. Representado na [Figura 11](#)

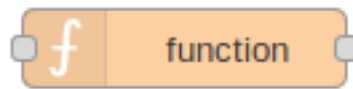
Figura 11 – Nó json



Fonte: Autor.

5. *function* - O nó Função permite que o código JavaScript seja executado nas mensagens que passam por ele. Foi utilizado neste trabalho para acessar cada variável individualmente e está representado na [Figura 12](#).

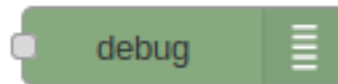
Figura 12 – Nó function



Fonte: Autor.

6. *debug* - O nó *Debug* é usado para exibir as mensagens na barra lateral no editor. Este nó foi utilizado para exibir as mensagens transmitidas e auxiliar na criação dos fluxos. Está representado na [Figura 13](#).

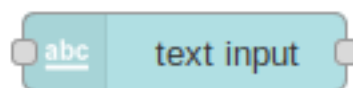
Figura 13 – Nó debug



Fonte: Autor.

7. *text* - Nó nativo que permite exibir um rótulo e/ou um valor no *dashboard*. Exibe os valores de taxa de crescimento e diâmetro médio no *dashboard*. Representado na [Figura 14](#).

Figura 14 – Nó text



Fonte: Autor.

8. *gauge* - Adiciona um ícone do tipo medidor à interface do usuário. Foi utilizado para exibir de forma mais visual o número de cogumelos identificados em cada cama de crescimento e está representado na [Figura 15](#).

Figura 15 – Nó gauge



Fonte: Autor.

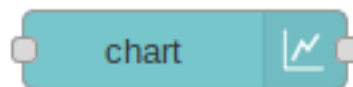
9. *LED* - Adiciona um ícone LED indicador de status ao *dashboard*. Indica se é o momento mais apropriado ou não para efetuar a colheita dos cogumelos. Representado na [Figura 16](#).

Figura 16 – Nó LED



Fonte: Autor.

10. *chart* - Nó que fornece gráficos para plotar variáveis. Usado neste trabalho para mostrar a quantidade de cogumelos que possuem diâmetros dentro de determinadas faixas. Representado na [Figura 17](#).

Figura 17 – Nó *chart*

Fonte: Autor.

5 RESULTADOS

Após a definição e teste do método de identificação dos cogumelos, o mesmo foi implementado utilizando os dados fotográficos obtidos no cultivo de cogumelos paris da propriedade no Espírito Santo.

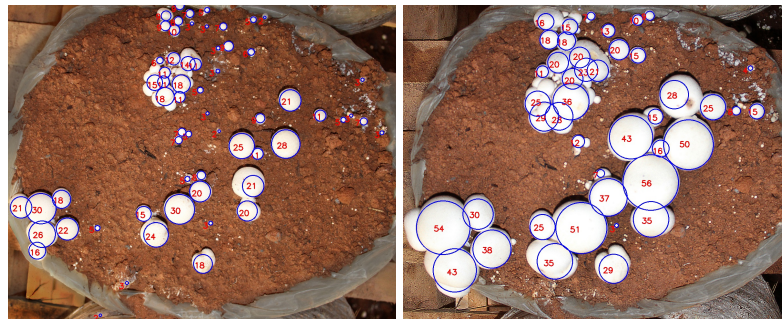
Nesta estapa, foram analisadas duas fases de crescimento com intervalo de 48 horas entre elas. Desta maneira, como o método de identificação se baseia em um conhecimento prévio das características dos objetos a serem identificados, foi necessário adicionar algumas linhas extras ao código utilizado durante os testes para que fossem estabelecidos parâmetros diferentes para cada fase. Após a definição destes parâmetros o código gerou como resultado as imagens apresentadas na [Figura 18](#).

A [Tabela 1](#) apresenta a precisão alcançada para cada cama de crescimento em relação ao número de cogumelos identificados e a determinação dos diâmetros dos mesmos. Também é apresentado a porcentagem de erro que corresponde à identificação de um mesmo cogumelo mais de uma vez, identificação de um objeto que não seja um cogumelo ou identificação de um cogumelo de uma cama de crescimento vizinha.

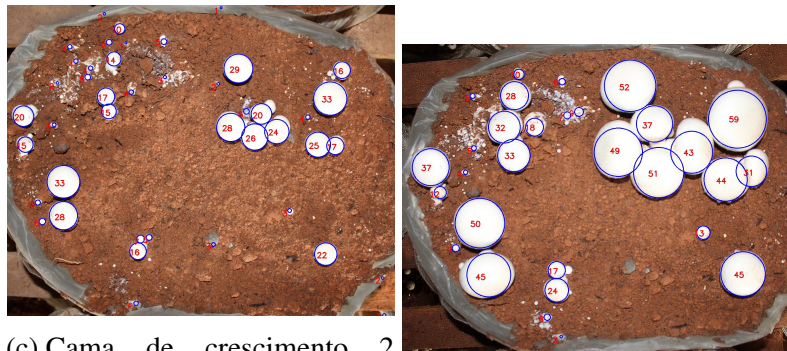
Tabela 1 – Precisão alcançada para cada cama de crescimento analisada

Cama de crescimento	Precisão de identificação	Precisão de diâmetro	Erro
1	59(89,4%)	51(77,3%)	3(4,5%)
2	39(97,5%)	33(82,5%)	5(2,5%)
3	37(80,4%)	30(65,2%)	1(2,0%)
4	71(75,5%)	62(66,0%)	1(1,0%)
5	37(84,1%)	33(75,0%)	2(4,5%)
6	30(96,8%)	28(90,3%)	1(3,2%)
7	35(97,2%)	28(77,8%)	1(2,8%)
8	45(70,3%)	38(59,4%)	5(7,8%)

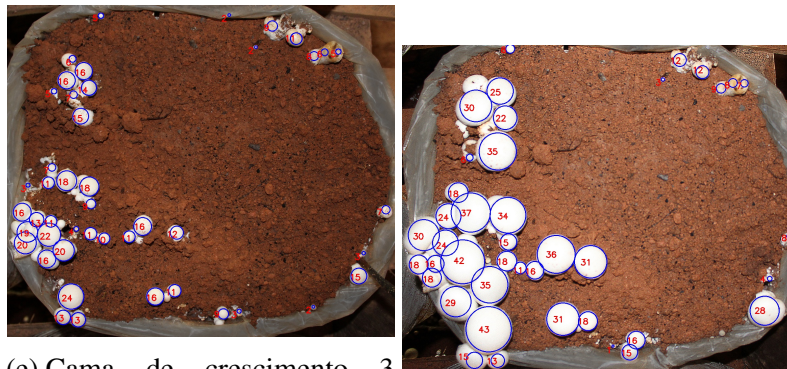
Foi alcançada uma precisão média de 86,4% na identificação dos cogumelos e 74,2% na determinação dos diâmetros dos mesmos. Os três cogumelos soltos presentes nas fotos (e) e (f) da [Figura 18](#) não foram considerados na estatística por estarem impróprios para o consumo e apresentar características diferentes dos padrões normais, o que impossibilita a identificação correta dos mesmos pelo sistema.



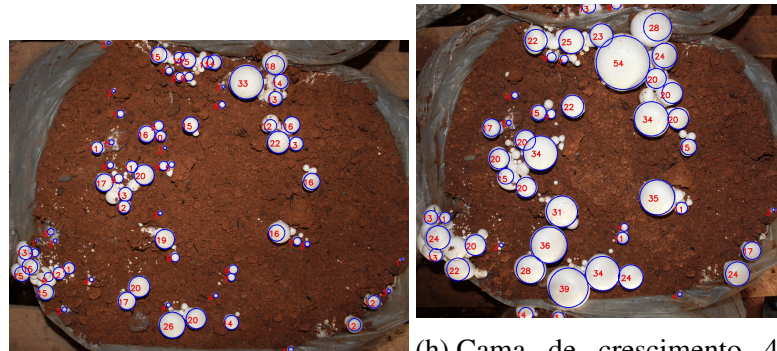
(a) Cama de crescimento 1 Fotografia obtida no dia 25/09/2019 (b) Cama de crescimento 1 Fotografia obtida no dia 27/09/2019



(c) Cama de crescimento 2 Fotografia obtida no dia 25/09/2019 (d) Cama de crescimento 2 Fotografia obtida no dia 27/09/2019



(e) Cama de crescimento 3 Fotografia obtida no dia 25/09/2019 (f) Cama de crescimento 3 Fotografia obtida no dia 27/09/2019

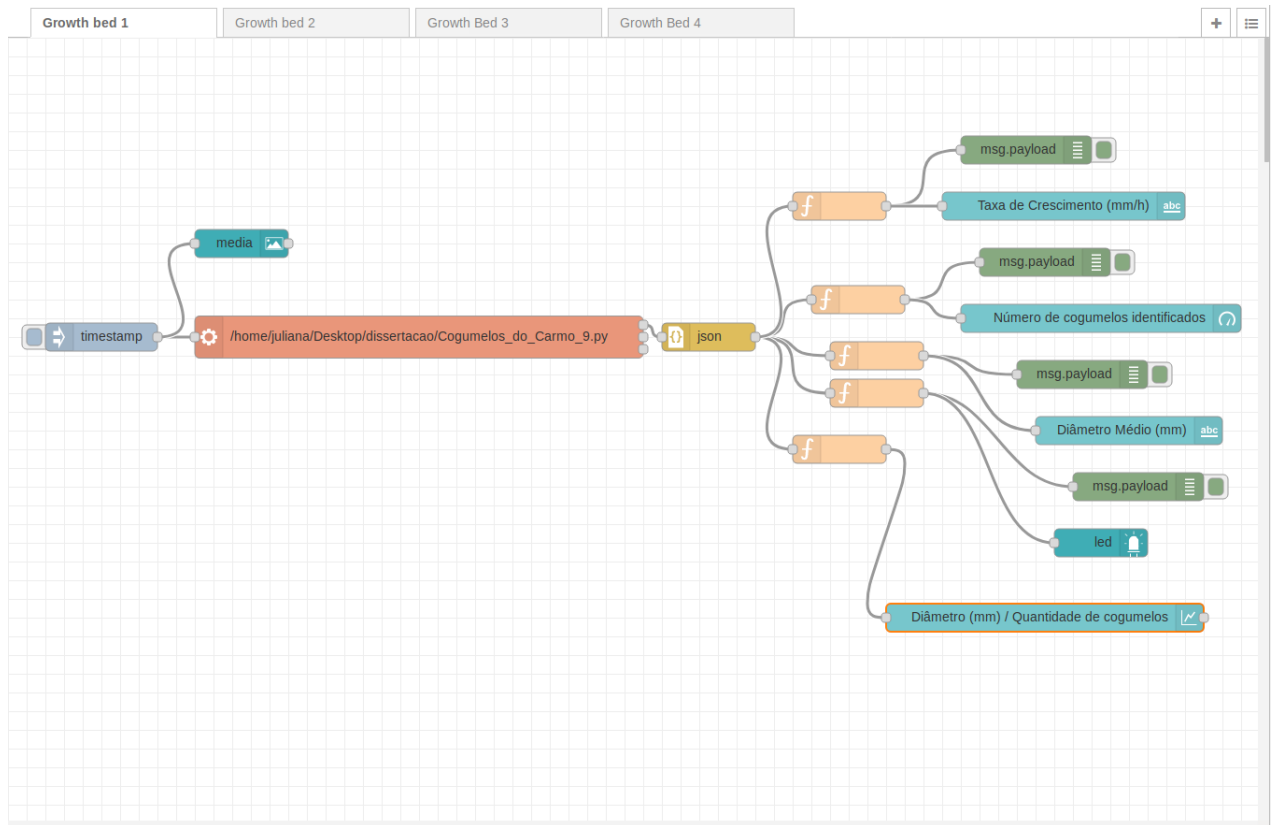


(g) Cama de crescimento 4 Fotografia obtida no dia 25/09/2019 (h) Cama de crescimento 4 Fotografia obtida no dia 27/09/2019

Figura 18 – Resultado da identificação dos cogumelos

Para cada cama de crescimento foi montado na ferramenta Node-Red um fluxo como o representado na [Figura 19](#).

Figura 19 – Fluxo criado no Node-Red para gerar a IHM

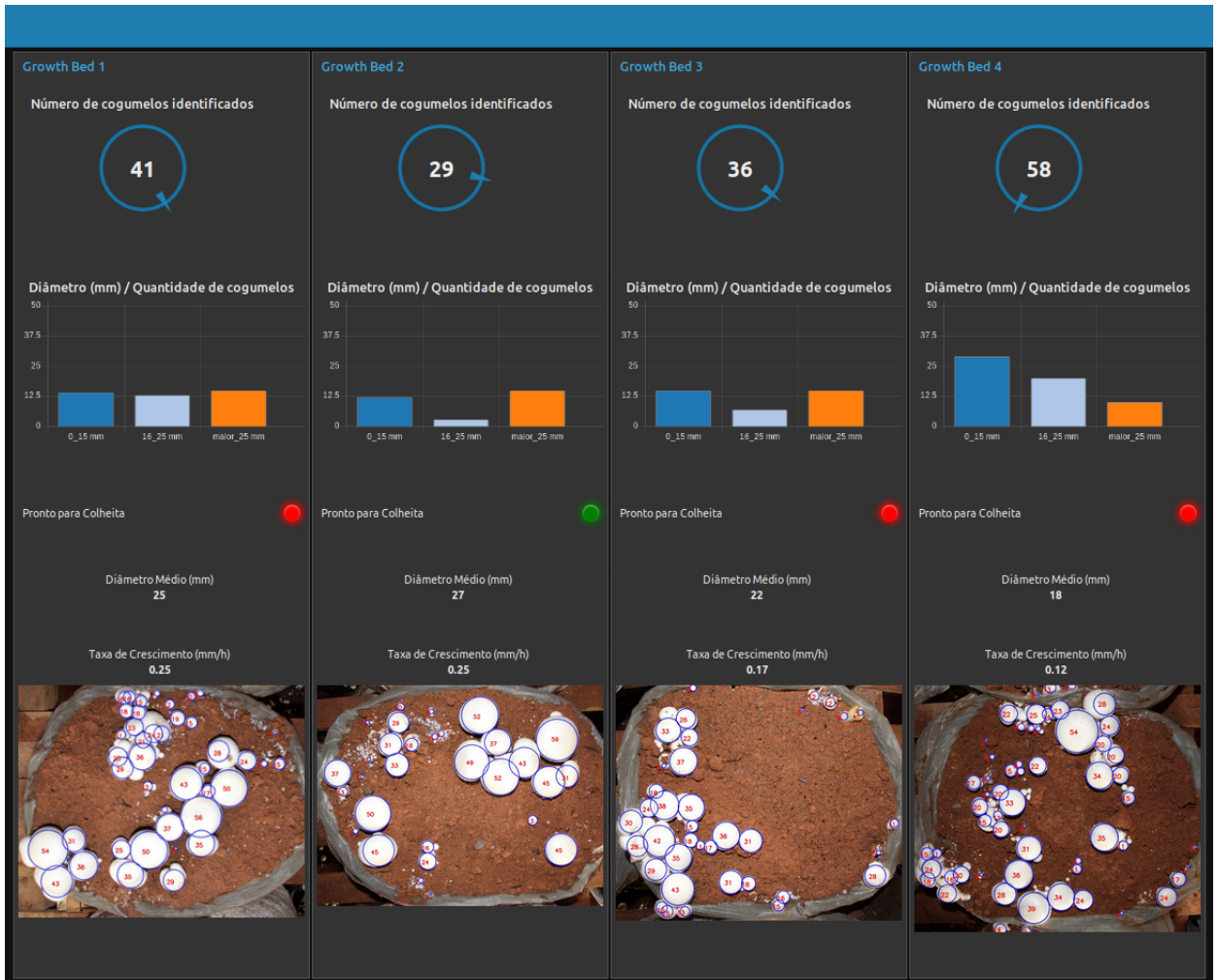


Fonte: Autor.

O fluxo é iniciado por um bloco *timestamp* que gera um evento fazendo com que o código endereçado no bloco *exec* (bloco laranja) seja executado e a fotografia da cama de crescimento analisada seja disponibilizada. O arquivo json resultante da execução do código é então convertido para um objeto (bloco json) a partir do qual as variáveis são acessadas individualmente através dos blocos do tipo *function*. Por fim, estas variáveis são enviadas para o *dashboard* para que possam ser visualizadas através de gráficos, gauges, de forma numérica, entre diversas outras possibilidades.

O resultado destes fluxos é um *dashboard* que transmite as informações de forma clara e visual pelo qual o produtor pode acompanhar o desenvolvimento do cultivo. As variáveis “número de cogumelos identificados”, “quantidade de cogumelos em cada intervalo de diâmetro”, “previsão de colheita” e “diâmetro médio” correspondem aos últimos dados fotográficos obtidos pelo sistema que são mostrados no final da página. Já a taxa de crescimento é calculada a partir da comparação com dados anteriores. A tela mostrada ao produtor quando o mesmo acessa o endereço disponibilizado pode ser vista na [Figura 20](#)

Figura 20 – IHM desenvolvida com a ferramenta Node-Red



Fonte: Autor.

6 CONCLUSÃO

Um sistema com visão computacional capaz de estimar o crescimento e determinar a previsão de colheita de cogumelos é proposto neste trabalho. O sistema identifica os cogumelos nas imagens e calcula automaticamente seus diâmetros e taxa de crescimento. Com base nestes dados, indica se é a hora mais apropriada ou não para efetuar a colheita.

O algoritmo foi desenvolvido utilizando a tecnologia python OpenCV e todo o sistema consiste basicamente de cinco etapas: aquisição de imagens, pré-processamento, segmentação, pós-processamento e análise.

A maior dificuldade para a identificação precisa dos cogumelos foi lidar com as características naturais da fungicultura que tornam o processamento de imagens uma tarefa desafiadora. A principal destas características é a forma como os cogumelos se desenvolvem. As aglomerações e sobreposições dificultam bastante a identificação dos mesmos nas imagens. Além disso, outros fatores como presença de terra na superfície, formato irregular e cogumelo com crescimento inclinado contribuem para aumentar o desafio.

Com o objetivo de superar estas barreiras, foram utilizadas técnicas aplicadas no processamento de imagens médicas, mais especificamente na identificação celular. As pesquisas nesta área já se encontram bastante desenvolvidas e muito do que é utilizado neste campo pode ser aplicado no processamento de imagens da fungicultura devido as semelhanças das características morfológicas.

As análises dos resultados mostram que o sistema proposto tem um bom desempenho em termos de identificação e medição de diâmetro de cogumelos. Foi alcançada uma precisão média de 86,4% na identificação dos cogumelos e 74,2% na determinação dos diâmetros dos mesmos.

Futuramente, é possível aperfeiçoar o método de identificação através da implementação de um sistema de reconhecimento utilizando inteligência artificial. O sistema proposto também pode ser combinado com sistemas de controle de variáveis como temperatura, umidade e gás carbônico para otimizar as condições de desenvolvimento dos cogumelos.

REFERÊNCIAS

AIKATERINI, C. *Farming Smartification*. Tese (Doutorado) — International Hellenic University, Thessaloniki – Greece, jan. 2016. Disponível em: <<https://repository.ihu.edu.gr/xmlui/bitstream/handle/11544/14456/Chioteli.Aikaterini.1106140003.Dissertation-signed.pdf?sequence=1>>. Citado na página 15.

ANTOLOVIC, D. Review of the Hough Transform Method, With an Implementation of the Fast Hough Variant for Line Detection. *Indiana University*, 2008. Citado na página 23.

ARAUJO, A. K.; BARBIRATO, J. M. R. C. Análise envoltória de dados (DEA): Um estudo bibliométrico sobre a produção científica internacional nos últimos dez anos (2006-2016). In: . JOINVILLE/SC - BRASIL: [s.n.], 2017. Disponível em: <<http://www.abepro.org.br/publicacoes/artigo.asp?e=enegep&a=2017&c=32993>>. Citado na página 51.

BHARGAVA, A.; BANSAL, A. Fruits and vegetables quality evaluation using computer vision: A review. *Journal of King Saud University - Computer and Information Sciences*, p. S131915781830209X, jun. 2018. ISSN 13191578. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S131915781830209X>>. Citado 4 vezes nas páginas 20, 26, 27 e 30.

BIENIEK, A.; MOGA, A. An efficient watershed algorithm based on connected components. *Pattern Recognition*, v. 33, n. 6, p. 907–916, jun. 2000. ISSN 00313203. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0031320399001545>>. Citado na página 22.

CHANDRASEKAR, L.; DURGA, G. Implementation of Hough Transform for image processing applications. In: *2014 International Conference on Communication and Signal Processing*. Melmaruvathur, India: IEEE, 2014. p. 843–847. ISBN 978-1-4799-3358-7 978-1-4799-3357-0 978-1-4799-3356-3. Disponível em: <<http://ieeexplore.ieee.org/document/6949962/>>. Citado na página 23.

CHEN, W.-L. et al. AgriTalk: IoT for Precision Soil Farming of Turmeric Cultivation. *IEEE Internet of Things Journal*, v. 6, n. 3, p. 5209–5223, jun. 2019. ISSN 2327-4662, 2372-2541. Disponível em: <<https://ieeexplore.ieee.org/document/8641389/>>. Citado na página 15.

CHOUHAN, S. S.; SINGH, U. P.; JAIN, S. Applications of Computer Vision in Plant Pathology: A Survey. *Archives of Computational Methods in Engineering*, fev. 2019. ISSN 1134-3060, 1886-1784. Disponível em: <<http://link.springer.com/10.1007/s11831-019-09324-0>>. Citado 2 vezes nas páginas 20 e 26.

CHUDASAMA, D. et al. Image Segmentation using Morphological Operations. *International Journal of Computer Applications*, v. 117, n. 18, p. 16–19, maio 2015. ISSN 09758887. Disponível em: <<http://research.ijcaonline.org/volume117/number18/pxc3903197.pdf>>. Citado na página 22.

COSTA, J. et al. Intelligent mushroom harvest prediction system proposal. In: *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*. Cáceres: IEEE, 2018. p. 1–7. ISBN 978-989-98434-8-6. Disponível em: <<https://ieeexplore.ieee.org/document/8399249/>>. Citado 4 vezes nas páginas 15, 16, 26 e 28.

- DING, Z.; ZHANG, R.; KAN, Z. Quality and Safety Inspection of Food and Agricultural Products by LabVIEW IMAQ Vision. *Food Analytical Methods*, v. 8, n. 2, p. 290–301, fev. 2015. ISSN 1936-9751, 1936-976X. Disponível em: <<http://link.springer.com/10.1007/s12161-014-9989-1>>. Citado na página 25.
- DOMÍNGUEZ, C.; HERAS, J.; PASCUAL, V. IJ-OpenCV: Combining ImageJ and OpenCV for processing images in biomedicine. *Computers in Biology and Medicine*, v. 84, p. 189–194, maio 2017. ISSN 00104825. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0010482517300823>>. Citado na página 25.
- FANG, R. et al. State of the World's Fungi 2018. *Royal Botanic Gardens, Kew*, p. 11, 2018. Citado na página 18.
- FELZENSZWALB, P. F.; HUTTENLOCHER, D. P. Distance Transforms of Sampled Functions. *Theory of Computing*, v. 8, n. 1, p. 415–428, 2012. ISSN 1557-2862. Disponível em: <<http://www.theoryofcomputing.org/articles/v008a019>>. Citado na página 33.
- FILHO, E. A. M.; CAVALINI, P. D. F. C. Plano de Negócio para Produção de Cogumelo Shimeji Desidratado. *Revista Perspectiva em Educação Gestão & Tecnologia*, v. 4, n. 7, p. 8, 2015. ISSN 2238 - 8486. Citado na página 16.
- G. Marota et al. Machine learning and data mining advance predictive big data analysis in precision animal agriculture. In: *Applications of data mining and prediction methods to animal sciences*. Baltimore, Maryland: Journal of Animal Science, 2017. Citado na página 16.
- GAO, S. Research on Medical Image Processing Method Based on the Matlab. In: DU, W. (Ed.). *Informatics and Management Science V*. London: Springer London, 2013. v. 208, p. 687–694. ISBN 978-1-4471-4795-4 978-1-4471-4796-1. Disponível em: <http://link.springer.com/10.1007/978-1-4471-4796-1_88>. Citado na página 24.
- GETTING the most out of published research - Scopus | Elsevier Solutions. Disponível em: <<https://www.elsevier.com/solutions/scopus/how-scopus-works>>. Citado na página 51.
- GILL, J.; GIRDHAR, A.; SINGH, T. A Review of Enhancement and Segmentation Techniques for Digital Images. *International Journal of Image and Graphics*, v. 19, n. 03, p. 1950013, jul. 2019. ISSN 0219-4678, 1793-6756. Disponível em: <<https://www.worldscientific.com/doi/abs/10.1142/S021946781950013X>>. Citado 2 vezes nas páginas 21 e 23.
- GLÄNZEL, W.; INTÉZET, M. T. A. K. *Bibliometrics as a research field: A course on theory and application of bibliometric indicators*. Magyar Tudományos Akadémia, Kutatásszervezési Intézet, 2003. Disponível em: <https://books.google.com.br/books?id=_TjMMgEACAAJ>. Citado na página 51.
- Paul V. C. Hough. *Method and Means for Recognizing Complex Patterns*. 1962. 3069654. Citado na página 23.
- K., A. R.; MORDVINTSEV, A. Google Summer of Code 2013 program, *OpenCV: Introduction to OpenCV-Python Tutorials*. 2013. Disponível em: <https://docs.opencv.org/master/d0/de3/tutorial_py_intro.html>. Citado na página 25.
- KAEHLER, A.; BRADSKI, G. *Learning OpenCV 3*. [S.l.]: O'Reilly Media, 2015. Citado na página 25.

KAUR, D.; KAUR, Y. Various Image Segmentation Techniques: A Review. p. 6, 2014. Citado na página 23.

KORNILOV, A.; SAFONOV, I. An Overview of Watershed Algorithm Implementations in Open Source Libraries. *Journal of Imaging*, v. 4, n. 10, p. 123, out. 2018. ISSN 2313-433X. Disponível em: <<http://www.mdpi.com/2313-433X/4/10/123>>. Citado na página 22.

LU, C.-P. et al. Development of a mushroom growth measurement system applying deep learning for image recognition. *Agronomy*, v. 9, n. 1, p. 32, 2019. ISSN 2073-4395. Disponível em: <<http://www.mdpi.com/2073-4395/9/1/32>>. Citado 4 vezes nas páginas 16, 19, 28 e 52.

MAHATA, K. et al. Fuzzy evaluated quantum cellular automata approach for watershed image analysis. In: *Quantum Inspired Computational Intelligence*. Elsevier, 2017. p. 259–284. ISBN 978-0-12-804409-4. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/B9780128044094000085>>. Citado na página 21.

MAHMOODI, M.; JAMES, L.; JOHANSEN, T. Automated advanced image processing for micromodel flow experiments; an application using labVIEW. *Journal of Petroleum Science and Engineering*, v. 167, p. 829–843, ago. 2018. ISSN 09204105. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0920410518301384>>. Citado na página 24.

MAHMUD, M. S. A. et al. Internet of things based smart environmental monitoring for mushroom cultivation. *Indonesian Journal of Electrical Engineering and Computer Science*, v. 10, n. 3, p. 847, 2018. ISSN 2502-4760, 2502-4752. Disponível em: <<http://www.iaescore.com/journals/index.php/IJEECS/article/view/11934>>. Citado na página 53.

MALDONADO, W.; BARBOSA, J. C. Automatic green fruit counting in orange trees using digital images. *Computers and Electronics in Agriculture*, v. 127, p. 572–581, set. 2016. ISSN 01681699. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0168169916305294>>. Citado na página 27.

MEIMBAN, R. J. et al. Blood Cells Counting using Python OpenCV. In: *2018 14th IEEE International Conference on Signal Processing (ICSP)*. Beijing, China: IEEE, 2018. p. 50–53. ISBN 978-1-5386-4673-1. Disponível em: <<https://ieeexplore.ieee.org/document/8652384/>>. Citado na página 30.

MORDVINTSEV, A.; K., A. *Image Thresholding — OpenCV-Python Tutorials 1 documentation*. 2013. Disponível em: <https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_thresholding/py_thresholding.html>. Citado na página 31.

MUSHROOMS. [S.l.]: National Agricultural Statistics Service (NASS), Agricultural Statistics Board, United States Department of Agriculture (USDA), 2019. Citado na página 18.

NELLUTLA, A. Image Processing Techniques Using LabVIEW. p. 6, 2018. Citado na página 25.

OTSU, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 9, n. 1, p. 62–66, jan. 1979. ISSN 0018-9472, 2168-2909. Disponível em: <<http://ieeexplore.ieee.org/document/4310076/>>. Citado na página 31.

PANAYI, E.; PETERS, G. W.; KYRIAKIDES, G. Statistical modelling for precision agriculture: A case study in optimal environmental schedules for *Agaricus Bisporus* production via variable domain functional regression. *PLOS ONE*, v. 12, n. 9, p. e0181921, set. 2017. ISSN 1932-6203. Disponível em: <<https://dx.plos.org/10.1371/journal.pone.0181921>>. Citado na página 18.

PATEL, K. K. et al. Machine vision system: a tool for quality inspection of food and agricultural products. *Journal of Food Science and Technology*, v. 49, n. 2, p. 123–141, abr. 2012. ISSN 0022-1155, 0975-8402. Disponível em: <<http://link.springer.com/10.1007/s13197-011-0321-4>>. Citado 2 vezes nas páginas 20 e 26.

POSADA-GOMEZ, R. et al. Digital Image Processing Using LabView. In: FO-LEA, S. (Ed.). *Practical Applications and Solutions Using LabVIEW®8482; Software*. InTech, 2011. ISBN 978-953-307-650-8. Disponível em: <<http://www.intechopen.com/books/practical-applications-and-solutions-using-labview-software/digital-image-processing-using-labview>>. Citado na página 24.

RADCLIFFE, J.; COX, J.; BULANON, D. M. Machine vision for orchard navigation. *Computers in Industry*, v. 98, p. 165–171, jun. 2018. ISSN 01663615. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0166361517305389>>. Citado na página 15.

S, P. R.; ARANGAIAH, V. M.; THANGARAJ, J. An Updated Anti-Collision System Between Machines in a Stack Yard Using Image Processing Techniques. In: *2018 3rd International Conference on Communication and Electronics Systems (ICCES)*. Coimbatore, India: IEEE, 2018. p. 109–113. ISBN 978-1-5386-4765-3. Disponível em: <<https://ieeexplore.ieee.org/document/8723896/>>. Citado na página 25.

SANTANA, I. Portal Embrapa (Versão 3.71.0) p03, *52º Curso sobre Cultivo de Cogumelos ensina a produzir em ambiente agroflorestal*. 2019. Disponível em: <<https://www.embrapa.br/busca-de-noticias/-/noticia/42679882/52-curso-sobre-cultivo-de-cogumelos-ensina-a-produzir-em-ambiente-agroflorestal>>. Citado na página 18.

SERT, E.; OKUMUS, I. T. Segmentation of Mushroom and Cap width Measurement using Modified K-Means Clustering Algorithm. *Advances in Electrical and Electronic Engineering*, v. 12, n. 4, p. 354 – 360, dez. 2014. ISSN 1804-3119, 1336-1376. Disponível em: <<http://advances.utc.sk/index.php/AEEE/article/view/1200>>. Citado na página 20.

SHARMA, R.; SHARMA, R. Image Segmentation using Morphological Operations for Automatic Region Growing. v. 4, p. 4, 2013. Citado 2 vezes nas páginas 22 e 32.

The MathWorks, Inc. *Image Processing Toolbox - MATLAB*. 1994. Disponível em: <<https://www.mathworks.com/products/image.html>>. Citado na página 24.

THOMAS, R. M.; JOHN, J. A review on cell detection and segmentation in microscopic images. In: *2017 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. Kollam, India: IEEE, 2017. p. 1–5. ISBN 978-1-5090-4967-7. Disponível em: <<http://ieeexplore.ieee.org/document/8074189/>>. Citado na página 29.

TILLET, R.; BATCHELOR, B. An algorithm for locating mushrooms in a growing bed. *Computers and Electronics in Agriculture*, v. 6, n. 3, p. 191–200, 1991. ISSN 01681699. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/016816999190001P>>. Citado na página 53.

- TOMARI, R. et al. RED BLOOD CELL COUNTING ANALYSIS BY CONSIDERING AN OVERLAPPING CONSTRAINT. v. 10, n. 3, p. 8, 2015. Citado 2 vezes nas páginas 32 e 33.
- WANG, F. et al. An automatic sorting system for fresh white button mushrooms based on image processing. *Computers and Electronics in Agriculture*, v. 151, p. 416–425, ago. 2018. ISSN 01681699. Disponível em: <<https://linkinghub.elsevier.com/retrieve/pii/S0168169918300309>>. Citado 2 vezes nas páginas 16 e 28.
- WANG, F. et al. Classification Method Research of Fresh Agaricus Bisporus Based on Image Processing. In: LI, D.; ZHAO, C. (Ed.). *Computer and Computing Technologies in Agriculture XI*. Cham: Springer International Publishing, 2019. v. 545, p. 333–340. ISBN 978-3-030-06136-4 978-3-030-06137-1. Disponível em: <http://link.springer.com/10.1007/978-3-030-06137-1_30>. Citado na página 28.
- XING, F.; YANG, L. Robust Nucleus/Cell Detection and Segmentation in Digital Pathology and Microscopy Images: A Comprehensive Review. *IEEE Reviews in Biomedical Engineering*, v. 9, p. 234–263, 2016. ISSN 1937-3333, 1941-1189. Disponível em: <<http://ieeexplore.ieee.org/document/7373572/>>. Citado 3 vezes nas páginas 21, 23 e 24.
- ZHENG, C.; AHMAD, K. The segmentation of images of biological cells: A survey of methods and systems. p. 26, 2008. Citado na página 29.
- ZUECH, N. *Understanding and Applying Machine Vision*. 2nd. ed. Vision Systems International Yardley, Pennsylvania: Marcel Dekker, 2000. Citado na página 26.
- ZUPIC, I.; ČATER, T. Bibliometric methods in management and organization. *Organizational Research Methods*, v. 18, n. 3, p. 429–472, 2015. ISSN 1094-4281, 1552-7425. Disponível em: <<http://journals.sagepub.com/doi/10.1177/1094428114562629>>. Citado na página 58.

Apêndices

APÊNDICE A – VISÃO COMPUTACIONAL NO CULTIVO DE COGUMELOS: UMA ANÁLISE BIBLIOMÉTRICA DA BASE DE CONHECIMENTO

A fim de realizar um levantamento sobre o uso da visão computacional na agricultura de precisão, mais especificamente no cultivo de cogumelos, o presente estudo utiliza-se da bibliometria como forma de verificar como estes temas têm sido tratados de forma conjunta na literatura acadêmica. Para isso são realizadas análise de cronologia da produção, identificação de autores, artigos e periódicos de destaque no tema e análises de citação, cocitação e coautoria.

A.1 Procedimentos metodológicos

Neste estudo foi realizado um levantamento da produção científica relacionada ao uso da visão computacional no cultivo de cogumelos. A fim de selecionar os artigos a serem utilizados na pesquisa, foi realizada uma busca na base de dados Scopus utilizando-se de palavras-chave do tema proposto e seus tesauros. Para isso, foram selecionadas as palavras-chave “computer vision”, “precision agriculture” e “mushroom cultivation” e a partir delas foram identificados seus tesauros “machine vision”, “precision farming”, “smart farming”, “mushroom production”, “mushroom farming” e “fungiculture”.

A busca foi refinada restringindo o tipo de documento. Optou-se pela utilização apenas de artigos científicos e a chave de busca utilizada para a consulta está apresentada na [Figura 21](#)

Figura 21 – Quadro da chave de busca utilizada na base Scopus

```
TITLE-ABS-KEY ( ( "computer vision" OR "machine vision" ) AND ( "precision
agriculture" OR "precision farming" OR "smart farming" ) AND ( "mushroom
cultivation" OR "mushroom production" OR "mushroom farming" OR "fungiculture" ) ) AND
DOCTYPE ( ar )
```

Fonte: Autor.

A base de dados Scopus foi escolhida por fornecer uma ampla cobertura da literatura científica e acadêmica com dados atuais e confiáveis. Seu conteúdo é composto por 24.600 títulos ativos e 5.000 editores rigorosamente controlados e selecionados por um conselho de revisão independente e usa uma rica arquitetura de metadados avançados para todo o conteúdo indexado (GETTING...).

Para a análise dos dados foram utilizados os softwares Microsoft Excel® para a geração de gráficos, tabelas e quadros e o VOSviewer para a construção dos mapas bibliométricos. As visualizações bibliométricas oferecem uma maneira poderosa de apresentar informações mais detalhadas em uma análise bibliométrica (GLÄNZEL; INTÉZET, 2003).

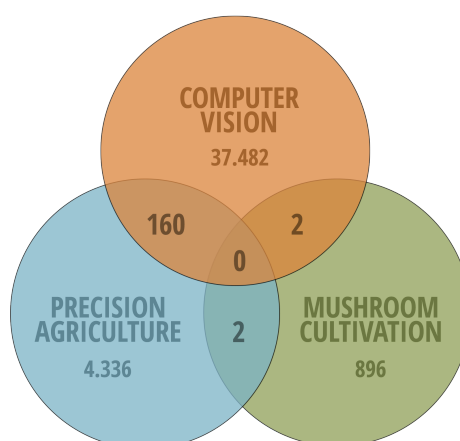
Para a elaboração desta pesquisa são realizadas as análises descritas a seguir:

1. Levantamento da cronologia da produção: esta análise consiste de um indicador de atividade científica que permite avaliar a evolução do tema em questão com relação ao número de publicações ao longo dos anos. Os dados são apresentados em formato gráfico, proporcionando melhor visualização dos períodos de maior produção.
2. Identificação dos autores com maior número de publicações: esta análise fornece informações sobre o número de publicações produzidas por um autor. Somente foram incluídos nesta análise autores com 3 ou mais publicações e autores e coautores foram tratados da mesma forma, contabilizando a pontuação por publicação seja ela como autor ou coautor do artigo.
3. Identificação dos periódicos de destaque no tema: tal análise permite inferir quais os periódicos que mais tem publicações no tema em questão e também a evolução temporal com relação ao número de publicações de cada um destes periódicos. Os dados são organizados em um quadro que apresenta o título do periódico e as respectivas quantidades de publicações e porcentagem equivalente em relação ao total de artigos analisados. São incluídos nesta análise os periódicos com duas ou mais publicações.
4. Identificação dos artigos com maior número de citações: esta análise busca identificar os artigos que se destacam na amostra em número de citações. Para este resultado os artigos foram classificados de forma descendente de acordo com o número de citações e foram selecionados aqueles que ocuparam as 10 primeiras posições.
5. Análises de citação, cocitação e coautoria: Estas análises permitem medir o impacto e a visibilidade de determinados autores, identificar a co-ocorrência de citação de dois autores na produção científica e usar dados de coautoria para medir colaboração, respectivamente (ARAUJO; BARBIRATO, 2017). Visualizações bibliométricas foram produzidas com auxílio do software VOSviewer.

A.2 Análise Bibliométrica

A amostra pesquisada corresponde aos artigos indexados na plataforma Scopus com consulta realizada no dia 6 de agosto de 2019. A busca foi efetuada utilizando as palavras-chave “computer vision”, “precision agriculture”, “mushroom cultivation” e seus tesouros. A fim de selecionar os documentos mais relevantes, a busca foi limitada apenas à artigos científicos e o resultado é mostrado na [Figura 22](#), que apresenta a quantidade de documentos retornados para cada uma das relações entre as palavras-chave e seus tesouros.

Figura 22 – Diagrama de Venn



Fonte: Autor.

Foram selecionados para análise mais aprofundada um total de 164 artigos que correspondem aos artigos retornados pela busca simultânea de 2 ou mais palavras-chave representados pelas interseções do diagrama de Venn. As Análises efetuadas utilizando esta seleção são apresentadas abaixo de acordo com a divisão proposta nos procedimentos metodológicos.

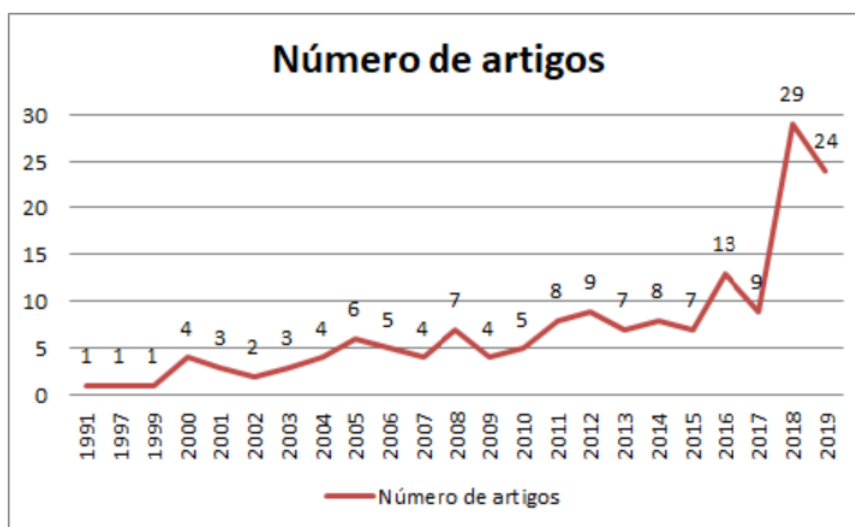
A.2.1 Levantamento da cronologia da produção

Na última década, os avanços em hardware e software para processamento digital de imagens motivaram vários estudos sobre a utilização da visão computacional como uma técnica potencial para a orientação ou controle de processos agrícolas e alimentares. A tecnologia de processamento de imagem tem sido amplamente utilizada em aplicações relacionadas ao cultivo de cogumelos, tais como: Development of a mushroom growth measurement system applying deep learning for image recognition (LU et al., 2019), Internet of things based smart

environmental monitoring for mushroom cultivation (MAHMUD et al., 2018) e An algorithm for locating mushrooms in a growing bed (TILLET; BATCHELOR, 1991).

A Figura 23 ilustra como se deu a distribuição da produção científica relacionada a este tema ao longo dos anos, permitindo observar em uma escala cronológica os períodos onde foram mais explorados.

Figura 23 – Distribuição da produção científica durante os anos



Fonte: Autor.

A análise do gráfico permite inferir que a relação entre os temas desta pesquisa passou a ser explorada com maior intensidade recentemente, obtendo a primeira publicação em 1991 e chegando ao máximo em 2018 com 29 publicações. É importante ressaltar que o número de publicações em 2019 apresenta grande possibilidade de superar a quantidade de 2018, devido à proximidade dos valores e o fato de a pesquisa ter sido realizada pouco mais de quatro meses antes do final do ano.

A.2.2 Identificação dos periódicos de destaque no tema

Foram encontrados 78 registros de periódicos indexados na base Scopus, que atendiam aos parâmetros da busca definida nos procedimentos metodológicos e nesta seção são apresentados os cinco periódicos com mais publicações.

A Tabela 2 apresenta os dados referentes à distribuição de registros quanto ao título do periódico e a porcentagem equivalente de artigos publicados em relação ao total identificado, correspondente a 164 artigos.

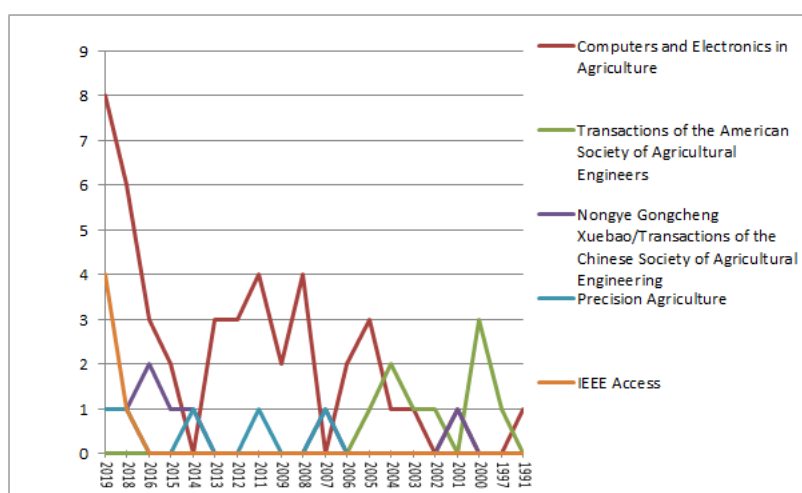
Tabela 2 – Número de artigos publicados por periódico

Periódico	Número de artigos	Porcentagem equivalente
Computers and Electronics in Agriculture.	44	26,83%
Transactions of the American Society of Agricultural Engineers	9	5,49%
Transactions of the Chinese Society of Agricultural Engineering	8	4,88%
Precision Agriculture	5	3,05%
IEEE Access	5	3,05%

Analisando a [Tabela 2](#), é possível identificar que o periódico Computers and Electronics in Agriculture representa, aproximadamente, 26% do total de artigos (164) encontrados na pesquisa. Apresentando uma diferença bastante significativa em relação ao segundo colocado (5,49%) e portanto se destacando como periódico de maior atuação na área estudada. Este deve ser monitorado pois tem mais possibilidade de apresentar artigos relevantes para os temas.

A [Figura 24](#), apresenta a distribuição do número de publicações com os temas em questão ao longo dos anos para os cinco periódicos de maior destaque na área.

Figura 24 – Distribuição do número de publicações com os temas em questão ao longo dos anos



Fonte: Autor.

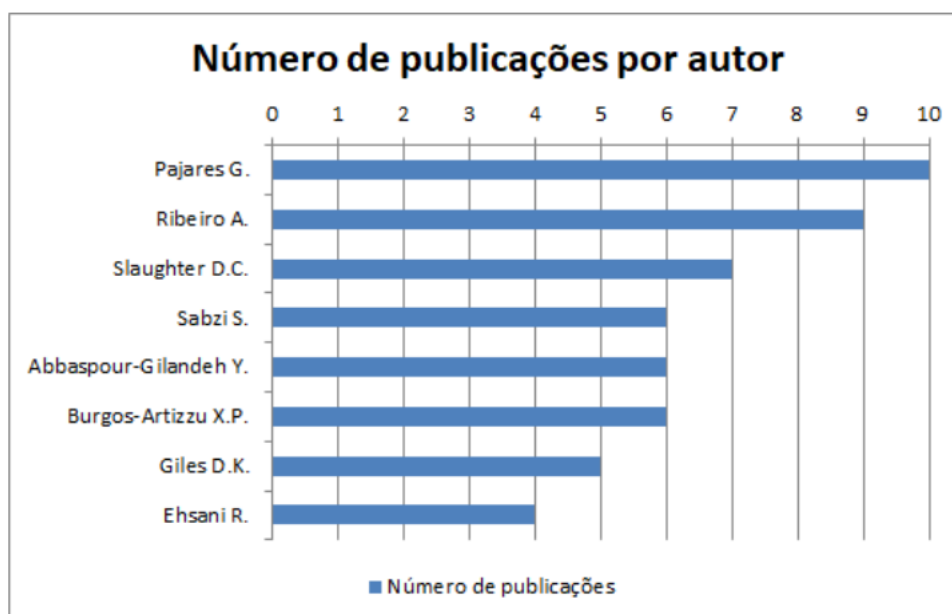
É possível inferir que o interesse do periódico *Computers and Electronics in Agriculture* no estudo do uso da visão computacional aplicada à agricultura começou em 1991 com a publicação do primeiro artigo no tema e cresceu significativamente nos últimos anos. Também é possível concluir que o tema perdeu força no periódico *Transactions of the American Society of Agricultural Engineers* que publicou seu último artigo na área no ano de 2005. Além de deixar claro o recente interesse do periódico *IEEE Access*, com 80% dos artigos relacionados aos temas publicados em 2019.

A.2.3 Identificação dos autores com maior número de publicações

A presente análise tem como objetivo identificar os autores que apresentaram o maior número de publicações no tema. Nesta pesquisa tratou-se da mesma forma a autoria e a co-autoria e a busca retornou um total de 538 autores e coautores.

A [Figura 25](#) autoref apresenta a distribuição de registros quanto à autoria dos artigos. Esta limita a apresentação dos autores que têm pelo menos quatro artigos indexados na base.

Figura 25 – Número de publicações por autor



Fonte: Autor.

Os nomes que mais se destacaram dentro deste contexto foram Pajares G., Ribeiro A. e Slaughter D. C. com 10, 9 e 7 publicações, respectivamente. Esses artigos foram publicados, geralmente, em conjunto com outros autores, inclusive alguns deles constituem uma colaboração entre os próprios autores em destaque. Esta informação pode ser comprovada na [Tabela 3](#) que apresenta os artigos mais citados de cada um desses autores.

Tabela 3 – Autores com o maior número de publicações e seus artigos mais citados

Autor	Artigo	Ano	Número de citações
Burgos-Artizzu X.P., Ribeiro A., Guijarro M., Pajares G.	Real-time image processing for crop/ weed discrimination in maize fields.	2011	127
Burgos-Artizzu X.P., Ribeiro A. , Guijarro M., Pajares G.	Real-time image processing for crop/ weed discrimination in maize fields.	2011	127
Slaughter D.C. , Giles D.K., Downey D.	Autonomous robotic weed control systems: A review.	2008	361
Sabzi S. , Abbaspour-Gilandeh Y., García-Mateos G.	A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms.	2018	6
Sabzi S., Abbaspour-Gilandeh Y. , García-Mateos G.	A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms.	2018	6
Burgos-Artizzu X.P. , Ribeiro A., Guijarro M., Pajares G.	Real-time image processing for crop/ weed discrimination in maize fields.	2011	127
Slaughter D.C., Giles D.K. , Downey D.	Autonomous robotic weed control systems: A review.	2008	361
Lee W.S., Ehsani R.	Sensing systems for precision agriculture in Florida.	2015	18

A análise da [Tabela 3](#) permite confirmar a representação destes autores no tema, já que seus artigos mais citados apresentam grande número de citações. O menor número de citações dos autores Sabzi S., Abbaspour-Gilandeh Y. e Ehsani R. pode ser explicado pelo fato de seus artigos terem sido publicados em datas mais recentes.

A.2.4 Artigos com maior número de citações

A [Tabela 4](#) apresenta os artigos que ocupam as dez primeiras posições na lista de artigos com maior número de citações.

Tabela 4 – Artigos mais citados

	Article Title	Authors	Journal	Citations
1	Autonomous robotic weed control systems: A review	Slaughter D.C., Giles D.K., Downey D.	Computers and Electronics Agriculture	361
2	Verification of color vegetation indices for automated crop imaging applications	Meyer G.E., Neto J.C.	Computers and Electronics Agriculture	275
3	Plant species identification using Elliptic Fourier leaf shape analysis	Neto J.C., Meyer G.E., Jones D.D., Samal A.K.	Computers and Electronics Agriculture	184
4	Real-time image processing for crop/weed discrimination in maize fields	Burgos-Artizzu X.P., Ribeiro A., Guijarro M., Pajares G.	Computers and Electronics Agriculture	127
5	Weed detection for site-specific weed management: Mapping and real-time approaches	López-Granados F.	Weed Research	119
6	A vision-based method for weeds identification through the Bayesian decision theory	Tellaeché A., Burgos-Artizzu X.P., Pajares G., Ribeiro A.	Pattern Recognition	93
7	Estimation of mango crop yield using image analysis - Segmentation method	Payne A.B., Walsh K.B., Subedi P.P., Jarvis D.	Computers and Electronics Agriculture	89
8	A new vision-based approach to differential spraying in precision agriculture	Tellaeché A., Burgos-Artizzu X.P., Pajares G., Ribeiro A., Fernández-Quintanilla C.	Computers and Electronics Agriculture	86
9	Precision weed control system for cotton	Lamm R.D., Slaughter D.C., Giles D.K.	Transactions of the American Society of Agricultural Engineers	83
10	Classification of weed species using color texture features and discriminant analysis	Burks T.F., Shearer S.A., Payne F.A.	Transactions of the American Society of Agricultural Engineers	82
	A computer vision approach for weeds identification through Support Vector Machines	Tellaeché A., Pajares G., Burgos-Artizzu X.P., Ribeiro A.	Applied Soft Computing Journal	82

Pode-se perceber que dos onze artigos presentes no quadro, seis foram publicados no periódico *Computers and Electronics in Agriculture*, o que comprova a relevância do mesmo na área. Além disso, também aparece na lista dos mais citados dois artigos publicados no periódico *Transactions of the American Society of Agricultural Engineers* que foi segundo colocado na classificação dos periódicos de maior destaque.

Vale também ressaltar a presença de diversos autores identificados como os nomes que mais se destacaram dentro do contexto da pesquisa.

A.2.5 Análises de citação, cocitação e coautoria

As citações são usadas como uma forma de medir a influência de estudos, autores ou periódicos. Se um artigo é muito citado, é considerado relevante. Esta proposta baseia-se no pressuposto de que os autores citam documentos que consideram importantes para o seu trabalho (ZUPIC; ČATER, 2015).

Na Figura 26 é apresentada a rede de citações, na qual cada grupo, representado por cores diferentes, é formado por um conjunto de autores que citaram o trabalho um do outro com mais frequência. Os membros de cada um destes grupos compartilham, em um grau mais elevado, idéias intelectuais semelhantes.

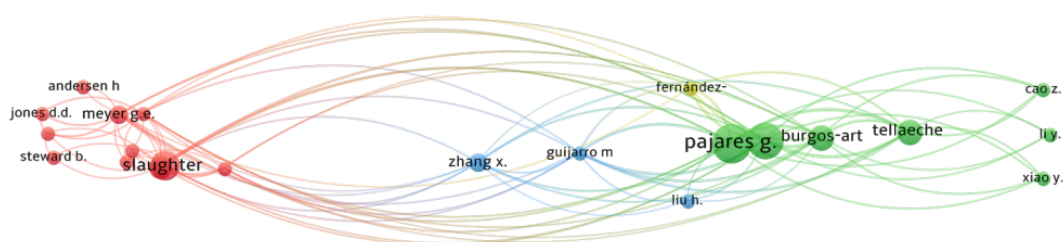


Figura 26 – Rede de Citação. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de documentos de um autor: 3. Número mínimo de citações: 20.

Os autores Pajares G., Slaughter D., Ribeiro A., Burgos-Artizzu e Tellaache A., representados pelos maiores círculos, foram citados mais vezes pelos demais autores da rede e representam os nomes de maior relevância. Alguns nomes como o de Ribeiro A. são omitidos na figura apenas por questão de visualização para evitar a sobreposição de autores.

Como nós centrais dentro da rede da ??, Pajares G., Guijarro M., Zhang X., Ribeiro A. e Burgos-Artizzu aparecem fortemente conectados com todos os outros grupos, indicando a contribuição dos mesmos para diferentes sub-áreas dentro do tema analisado.

Outra informação importante sobre os autores é o grau de co-citação. A análise de co-citação representada na [Figura 27](#), define a frequência com que dois autores são citados juntos. Quanto maior a incidência de cocitação, mais provável é que seus trabalhos estejam relacionados.

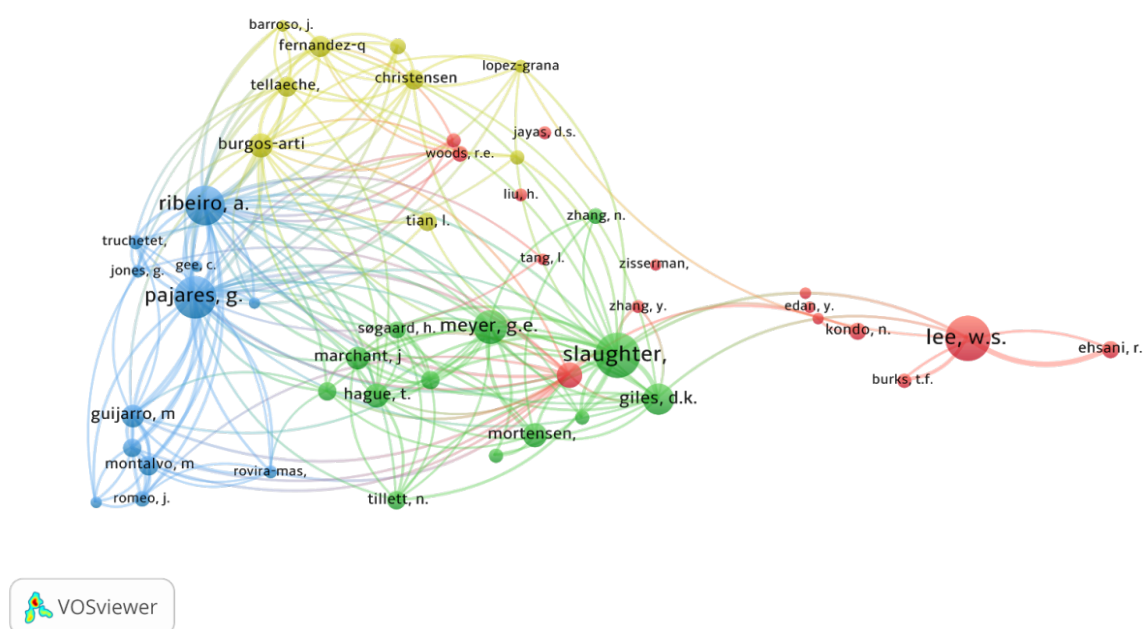


Figura 27 – Rede de Cocitação. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de citações: 20.

Quatro agrupamentos (azul, amarelo, verde e vermelho) aparecem como os principais campos. O grupo azul, refere-se aos autores interessados em assuntos relacionados à distinção entre mudas e ervas daninhas, aplicação localizada de herbicidas e veículos agrícolas, incluindo trabalhos de autores como Pajares G., Ribeiro A. e Guijarro M.. Os grupos amarelo e verde apresentam interesses bastante semelhantes aos do grupo azul como pode-se perceber na figura X devido à proximidade entre os círculos que representam os autores. Para o grupamento amarelo composto por Burgos-Artizzu, López-Granados, Fernández-Quintanilla, Tellaache, e outros pode-se adicionar à lista de interesses segmentação de vegetação e colheita robótica. Já para o grupo verde de Slaughter D, Marchant J., Giles D., Meyer G. e outros pode-se incluir os temas reconhecimento de imagem aplicado à agropecuária e identificação de espécies de plantas. Por fim, o grupo vermelho refere-se aos trabalhos dentro de perspectivas como sistemas de sensores,

estimação de massa de frutas e detecção de frutas caídas. Entre os autores com publicações nestes assuntos estão Lee W.S., Zhang Q., Ehsani R. e Zisserman A..

A análise de coautoria fornece evidências de colaboração e produz a estrutura social do campo de estudo. Ou seja, através dela é possível identificar autores que trabalham juntos na elaboração de artigos e a frequência com que colaboram uns com os outros.

Na Figura 28 é apresentada a rede de coautoria da produção científica sobre o uso da visão computacional na agricultura de precisão.

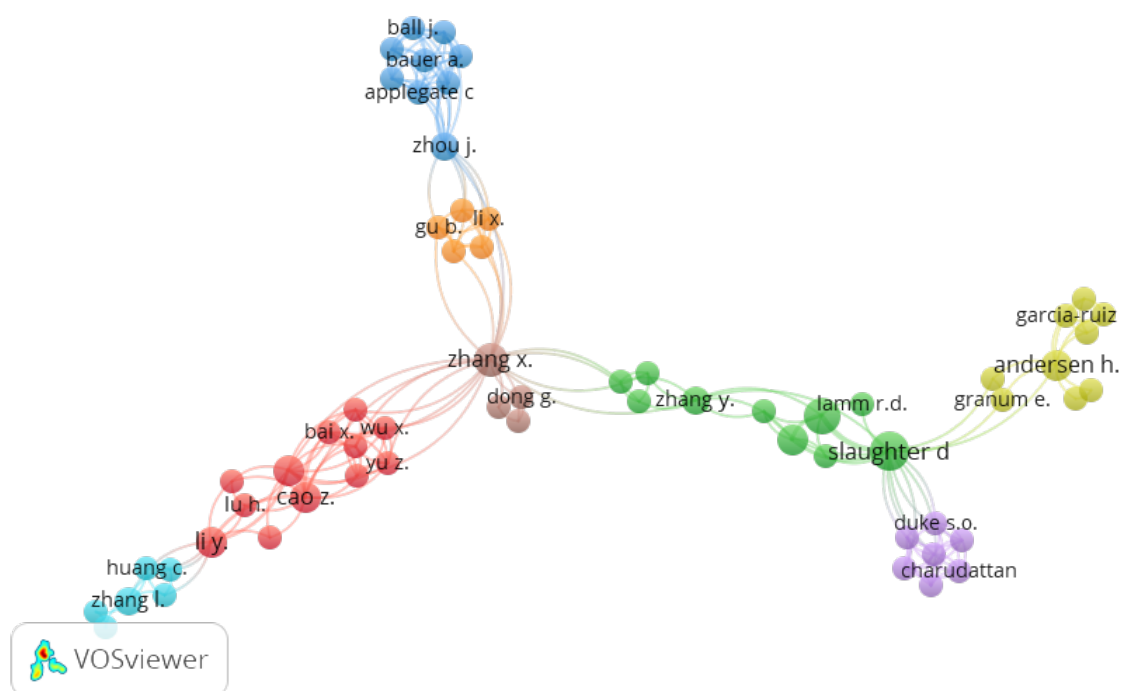


Figura 28 – Rede de coautoria. Fonte: Visualização no VOSviewer com dados do Scopus. Número mínimo de documentos para um autor: 1. Número mínimo de citações por autor: 0.

Na ??, cada autor é representado por um nó cujo tamanho é proporcional ao número de artigos publicados pelos mesmos. Os autores Slaughter D., Downey D., Giles D. e Zhang X. são representados por nós grandes, indicando sua significativa produtividade. É importante ressaltar que foram incluídos nesta análise somente os itens que apresentavam conexões uns com os outros e novamente alguns dos nomes dos autores são omitidos na imagem para evitar sobreposição. Essa rede de colaboração não apenas revela os pesquisadores mais produtivos, como também mostra através dos agrupamentos de cores diferentes os vários grupos de pesquisa criados pelas colaborações.

A.3 Considerações finais

Neste estudo foi realizado um levantamento da produção científica relacionada ao uso da visão computacional na agricultura de precisão, mais especificamente, no cultivo de cogumelos. Destaca-se o fato de que a pesquisa na amostra relacionando simultaneamente os conceitos “Computer Vision”, “Precision Agriculture” e “Mushroom Cultivation” não retornou nenhum trabalho, o que indica uma relação ainda não explorada na pesquisa científica.

Além disso, a análise da cronologia da produção permite inferir que a relação entre os temas passou a ser explorada com maior intensidade recentemente, com o número de publicações em 2018 atingindo mais que o dobro do número de publicações do ano com melhor resultado entre os anos anteriores e 2019 apresentando grande possibilidade de superar esta estatística, devido à proximidade dos valores e o fato de a pesquisa ter sido realizada no mês de agosto.

O periódico que mais se destacou na área foi o *Computers and Electronics in Agriculture*. O mesmo possui 44 artigos nos temas analisados, número quase 5 vezes maior que o segundo colocado, *Transactions of the American Society of Agricultural Engineers* que possui 9 artigos. Sua relevância é comprovada nas análises posteriores, já que vários dos autores com maior número de publicações elegeram o periódico para divulgar seus trabalhos e dos onze artigos com maior número de citações, seis foram publicados no periódico *Computers and Electronics in Agriculture*.

Também pode-se identificar com clareza os autores que mais se destacaram dentro deste contexto. Vários dos nomes citados na análise do número de publicações também aparecem na identificação dos artigos com maior número de citações, o que ratifica a importância destes autores para a área de estudo não só de maneira quantitativa mas também qualitativa. Desta forma artigos escritos por autores como Pajares G., Ribeiro A., Slaughter D. C. e Burgos-Artizzu X. P. devem ser considerados como referência quando se trata do tema em questão.

Através da análise de citação é possível perceber que os autores Pajares G., Slaughter D., Ribeiro A., Burgos-Artizzu e Tellaeché A., representam nomes de relevância quando o número de citações é o requisito em consideração. Já as análises de cocitação e coautoria revelam o significativo grau de colaboração entre os autores da coleção de publicações analisadas.

APÊNDICE B – CÓDIGO

```
#!/usr/bin/python3

## import the necessary packages
import cv2
import numpy as np
from matplotlib import pyplot as plt
from PIL import Image, ExifTags
from PIL.ExifTags import TAGS
import os,sys

#declare variables
output = [ ];
array_diameters = [ ];
number_mushrooms = [ ];
intervalo_1 = [ ];
intervalo_2 = [ ];
intervalo_3 = [ ];
Harvest_Time = [ ];
growth_rate = [ ];
avarage = 0;
harvest_time = 0;
j = 0;
y = 1;
f = 0;

# images path
path = '/home/juliana/Desktop/dissertacao/image_processing_backup'

for filename in sorted(os.listdir(path)):
    img = cv2.imread(os.path.join(path, filename))
    #resize
    im = cv2.resize(img, None, fx = 0.3, fy = 0.3, interpolation = cv2.INTER_CUBIC)

##Bilateral Filter
```

```
blur = cv2.bilateralFilter(im,5,130,130)

##increasing contrast
kernel = np.array([[0, -1, 0], [-1, 5, -1], [0, -1, 0]])
sharp = cv2.filter2D(blur, -1, kernel)
sharp_edges = np.concatenate((blur, sharp), axis = 1)

##extrair metadados
#função para extrair metadados da imagem
def get_exif(S1):
    image = Image.open(os.path.join(path, filename))
    image.verify()
    return image._getexif()
exif = get_exif('filename')
#função para mostrar metadados com etiquetas
def get_labeled_exif(exif):
    labeled = {}
    for (key, val) in exif.items():
        labeled[TAGS.get(key)] = val
    return labeled
exif = get_exif('filename')
labeled = get_labeled_exif(exif)
#função para extrair e mostrar apenas o valor de distância focal do metadados
def get_FocalLength(exif):
    if not exif:
        raise ValueError("No EXIF metadata found")
    FocalLength = {}
    for (idx, tag) in TAGS.items():
        if tag == 'FocalLength':
            FocalLength = exif[idx]
    return FocalLength
exif = get_exif('filename')
focallength = get_FocalLength(exif)
focallength = focallength[0]
```



```
## convert image to grayscale
gray = cv2.cvtColor(sharp, cv2.COLOR_BGR2GRAY)

## Otsu's thresholding
thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY | cv2.THRESH_OTSU)[1]
#ret1, thresh = cv2.threshold(gray,137,255,cv2.THRESH_BINARY)

##post processing
kernel = np.ones((2,2),np.uint8)

#image erosion
img_erosion = cv2.erode(thresh,kernel,iterations = 1)

#image dilation
img_dilation = cv2.dilate(img_erosion,kernel,iterations = 1)

#fill holes
#Get elliptical/circular shaped kernels. Pass the shape and size of the kernel.
kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (3, 3))
#Apply morphological closing
morph = cv2.morphologyEx(img_dilation , cv2.MORPH_CLOSE, kernel)

#Take the distance transform
dist = cv2.distanceTransform(morph,cv2.DIST_L2,cv2.DIST_MASK_PRECISE)
```

```

#Create a template.
if filename[0] == 'P':
    borderSize = 19
    distborder = cv2.copyMakeBorder(dist, borderSize, borderSize, borderSize, borderSize, cv2.BORDER_CONSTANT
| cv2.BORDER_ISOLATED, 0)
    gap = 4
    kernel2 = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (2*(borderSize-gap)+1, 2*(borderSize-
gap)+1))
    kernel2 = cv2.copyMakeBorder(kernel2, gap, gap, gap, gap, cv2.BORDER_CONSTANT | cv2.BORDER_ISOLATED
0)

#Take the template's distance transform and use it as the template
distTempl = cv2.distanceTransform(kernel2, cv2.DIST_L2,
cv2.DIST_MASK_PRECISE)

#Perform template matching
nxcor = cv2.matchTemplate(distborder, distTempl, cv2.TM_CCOEFF_NORMED)

#Find the local maxima of the resulting image. Location of the maxima correspond to circle centers
and max values correspond to their radii.
mn, mx, _, _ = cv2.minMaxLoc(nxcor)
th, peaks = cv2.threshold(nxcor, mx*0.58, 255, cv2.THRESH_BINARY)
peaks8u = cv2.convertScaleAbs(peaks)

#Extract the peak location and its strength
_, contours, hierarchy = cv2.findContours(peaks8u, cv2.RETR_CCOMP,
cv2.CHAIN_APPROX_SIMPLE)
peaks8u = cv2.convertScaleAbs(peaks) #to use as mask

#Create a template.
if filename[0] == 'S':
    borderSize = 25
    distborder = cv2.copyMakeBorder(dist, borderSize, borderSize, borderSize, borderSize, cv2.BORDER_CONSTANT
| cv2.BORDER_ISOLATED, 0)
    gap = 6
    kernel2 = cv2.getStructuringElement(cv2.MORPH_ELLIPSE,
(2*(borderSize-gap)+1, 2*(borderSize-gap)+1))
    kernel2 = cv2.copyMakeBorder(kernel2, gap, gap, gap, gap,

```

```

cv2.BORDER_CONSTANT | cv2.BORDER_ISOLATED, 0)

    #Take the template's distance transform and use it as the template
    distTempl = cv2.distanceTransform(kernel2, cv2.DIST_L2,
cv2.DIST_MASK_PRECISE)

    #Perform template matching
    nxcor = cv2.matchTemplate(distborder, distTempl, cv2.TM_CCOEFF_NORMED)

    #Find the local maxima of the resulting image. Location of the maxima correspond to circle centers
and max values correspond to their radii
    mn, mx, _, _ = cv2.minMaxLoc(nxcor)
    th, peaks = cv2.threshold(nxcor, mx*0.6, 255, cv2.THRESH_BINARY)
    peaks8u = cv2.convertScaleAbs(peaks)

    #Extract the peak location and its strength
    _, contours, hierarchy = cv2.findContours(peaks8u, cv2.RETR_CCOMP,
cv2.CHAIN_APPROX_SIMPLE)
    peaks8u = cv2.convertScaleAbs(peaks) #to use as mask

intervalo1 = 0;
intervalo2 = 0;
intervalo3 = 0;
ht = 0;
add_diameters = 0;

for i in range(len(contours)):
    x, y, w, h = cv2.boundingRect(contours[i])
    _, mx, _, mxloc = cv2.minMaxLoc(dist[y:y+h, x:x+w],
peaks8u[y:y+h, x:x+w])
    cv2.circle(im, (int(mxloc[0]+x), int(mxloc[1]+y)), int(mx), (255, 0, 0), 2)
    num_mushroom = i

    #distance to object (mm) = (focal length (mm) * real height of the object (mm) * image height (pixels))/
(object height (pixels) * sensor height (mm))

    # write mushroom diameter
    real_diameter = ((int(mx)*2)*14.9*383)/(focallength*972)
    cv2.putText(im, "{}".format(int(real_diameter)), (int(x) - 20, int(y) + 20),
cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 0, 255), 2)

```

```
add_diameters = (add_diameters + real_diameter)
#create histogram of diameters values
if real_diameter < 16:
    intervalo1 = intervalo1 + 1
elif real_diameter > 15 and real_diameter < 26:
    intervalo2 = intervalo2 + 1
else:
    intervalo3 = intervalo3 + 1

#Harvest estimation
if real_diameter > 39:
    harvest_time = harvest_time + 1

number_mushrooms.append(num_mushroom)
intervalo_1.append(intervalo1)
intervalo_2.append(intervalo2)
intervalo_3.append(intervalo3)

#Harvest estimation (if number of mushrooms bigger than 39mm it's higher than 50% of total than it's
harvest time )
if ((harvest_time*100)/num_mushroom) > 50:
    ht = 1
    Harvest_Time.append(ht)

#calculate avarage diameter value
avarage = add_diameters/num_mushroom

#save add_diametersvalues in an array
array_diameters.append(int(avarage))

j += 1

os.chdir('/home/juliana/Desktop/dissertacao/results')
#Filename
filename = "result_" + str(j) + ".jpg"
# Using cv2.imwrite() method
# Saving the image
```

```

cv2.imwrite(filename, im)

#calculate growth rate

#growth rate =>  $G = \frac{S_n - S_1}{T}$  => where S1 is the first measurement result, Sn is the nth measurement
result, and T is the time period from S1 to Sn.

for k in range(int(len(array_diameters)/2)):

    growth_rate.append(round(((array_diameters[(int(len(array_diameters)/2)+y)] - array_diameters[k])/48), 2))

    y += 1

    k += 1

#create Json output

output = "{\"numberMushrooms1\": " + str(number_mushrooms[4]) + ", \"numberMushrooms2\": " +
str(number_mushrooms[5]) + ", \"numberMushrooms3\": " +
str(number_mushrooms[6]) + ", \"numberMushrooms4\": " + str(number_mushrooms[7]) + ", \"avarageDi-
iameter1\": " + str(array_diameters[4]) + ", \"avarageDiameter2\": " + str(array_diameters[5]) + ", \"avarageDia-
iameter3\": " + str(array_diameters[6]) + ", \"avarageDiameter4\": " + str(array_diameters[7]) + ", \"intervalo1\": " +
str(intervalo_1[4]) + ", \"intervalo2\": " + str(intervalo_2[4]) + ", \"intervalo3\": " + str(intervalo_3[4]) +
", \"intervalo21\": " + str(intervalo_1[5]) + ", \"intervalo22\": " + str(intervalo_2[5]) + ", \"intervalo23\": " +
str(intervalo_3[5]) + ", \"intervalo31\": " + str(intervalo_1[6]) + ", \"intervalo32\": " + str(intervalo_2[6]) +
", \"intervalo33\": " + str(intervalo_3[6]) + ", \"intervalo41\": " + str(intervalo_1[7]) + ", \"intervalo42\": " +
str(intervalo_2[7]) + ", \"intervalo43\": " + str(intervalo_3[7]) + ", \"HarvestTime1\": " + str(Harvest_Time[4]) +
", \"HarvestTime2\": " + str(Harvest_Time[5]) + ", \"HarvestTime3\": " + str(Harvest_Time[6])
+ ", \"HarvestTime4\": " + str(Harvest_Time[7]) + ", \"growthrate1\": " + str(growth_rate[0]) + ", \"growthrate2\": " +
str(growth_rate[1]) + ", \"growthrate3\": " + str(growth_rate[2]) + ", \"growthrate4\": " + str(growth_rate[3]) +
"}"

print(output)

```