



**UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ**  
**CAMPUS LUIZ MENEGHEL - CENTRO DE CIÊNCIAS**  
**TECNOLÓGICAS**  
**BACHARELADO EM SISTEMAS DE INFORMAÇÃO E**  
**LICENCIATURA EM COMPUTAÇÃO**

**BRUNO ALEXANDER GUENTA TSURUSHIMA**

**MÉTODO PARA DETERMINAÇÃO DE**  
**TAMANHOS DE GOTAS EM IMAGENS**

Bandeirantes

2019

**BRUNO ALEXANDER GUENTA TSURUSHIMA**

**MÉTODO PARA DETERMINAÇÃO DE  
TAMANHOS DE GOTAS EM IMAGENS**

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação e Licenciado em Computação.

Orientador: Prof. Me. Wellington A. Della Mura

Coorientador: Prof. Dr. Rone B. de Oliveira

Bandeirantes

2019

BRUNO ALEXANDER GUENTA TSURUSHIMA

# MÉTODO PARA DETERMINAÇÃO DE TAMANHOS DE GOTAS EM IMAGENS

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação e Licenciado em Computação.

## COMISSÃO EXAMINADORA

---

Prof. Me. Wellington Aparecido Della Mura  
UENP – *Campus* Luiz Meneghel

---

Prof. Dr. Bruno Miguel Nogueira de Souza  
UENP – *Campus* Luiz Meneghel

---

Prof. Dr. Ederson Marcos Sgarbi  
UENP – *Campus* Luiz Meneghel

Bandeirantes, \_\_\_\_ de \_\_\_\_\_ de 2019

## RESUMO

Neste trabalho é apresentado um método para determinação de tamanho de gotas em imagens, utilizando uma câmera *ultra slow-motion*, determinação de configurações de câmera (resolução, abertura de lente e tempo de exposição), técnicas de processamento de imagens e algoritmo de aprendizado de máquina. A validação do método foi realizada comparando os resultados de tamanhos de gotas do método proposto com os resultados do *software* de mensuração de objetos ISCapture 2.2.1 utilizando a mesma imagem de gotas.

**Palavras-chave:** Visão Computacional, Processamento de imagens digitais, Aprendizado de Máquina, Tecnologia de Aplicação.

## **ABSTRACT**

This research presents a method for determination of droplet size in images, using an ultra slow-motion camera, determination of camera configurations (resolution, lens aperture and exposure time), image processing techniques and machine learning algorithm. The validation of this method was performed by comparing the results of droplet sizes of the proposed method with the results of the ISCapture 2.2.1 measure software using the same droplets image.

**Keywords:** Computer Vision, Digital Image Processing, Machine Learning, Application Technology.

## LISTA DE FIGURAS

Figura 1 – Etapas de processamento de imagens (Vulgarakis et al., 2016). .....	8
Figura 2 – Fluxograma estudado por Chen et al. (2013) .....	9
Figura 3. Resumo geral das etapas do processamento de imagens.....	10
Figura 5 - Ilustração de profundidade de campo de abertura 2.8. ....	12
Figura 6 - Ilustração de profundidade de campo de abertura 5.6. ....	13
Figura 7 – Imagens de gotas com diferentes configurações de abertura de lente. (a) Abertura de lente 2.8. (b) Abertura de lente 5.6. ....	13
Figura 9. Imagem das gotas sem filtro de suavização. ....	15
Figura 10. Imagem das gotas com filtro de suavização Mediana. ....	15
Figura 11. Imagem das gotas com filtro de suavização Gaussiana. ....	16
Figura 12. Imagem em estado original utilizada para testes. ....	17
Figura 13. Imagens utilizando Prewitt sem suavização. (a) Prewitt Horizontal (b) Prewitt Vertical.....	18
Figura 14. Imagens utilizando Prewitt com suavização Mediana. (a) Prewitt Horizontal (b) Prewitt Vertical.....	19
Figura 15. Imagens utilizando Prewitt com suavização Gaussiana. (a) Prewitt Horizontal (b) Prewitt Vertical.....	19
Figura 16. Imagens utilizando Sobel sem suavização. (a) Filtro Sobel Horizontal (b) Filtro Sobel Vertical .....	20
Figura 17. Imagens utilizando Sobel com suavização Mediana. (a) Filtro Sobel Vertical (b) Filtro Sobel Horizontal .....	21
Figura 18. Imagens utilizando Sobel com suavização Gaussiana. (a) Sobel Horizontal (b) Sobel Vertical.....	21
Figura 19. Imagem utilizando Laplaciano sem filtro de suavização. ....	22
Figura 20. Imagem com Laplaciano com filtro de suavização Mediana. ....	23
Figura 21. Imagem com Laplaciano utilizando suavização Gaussiana. ....	23
Figura 22. Imagem com <i>Canny Edge</i> utilizando suavização Gaussiana. (a) Imagem original. (b) Imagem binária resultante do <i>Canny Edge</i> . (c) Imagem original com bordas detectadas na imagem binária em vermelho. ....	24
Figura 23. Testes com valores para <i>Canny Edge</i> . (a) Imagem aplicando <i>Canny Edge</i> com valores 0 e 50 para mínimo e máximo. (b) Imagem aplicando <i>Canny Edge</i> com valores 50 e 100 para mínimo e máximo. (c) Imagem aplicando <i>Canny Edge</i> com valores 100 e 150 para	

mínimo e máximo. (d) Imagem aplicando *Canny Edge* com valores 150 e 200 para mínimo e máximo. (e) Imagem aplicando *Canny Edge* com valores 200 e 250 para mínimo e máximo.25

Figura 24. Imagens de gotas utilizadas como exemplo para mostrar tons de cinza. (a) Gota Focada (b) Gota Desfocada.....	26
Figura 25. Média dos tons de cinza de gota focada e gota desfocada. ....	27
Figura 26. Gráfico 3D dos tons de cinza de uma gota focada. ....	27
Figura 28. Imagem das bordas no interior de uma gota.....	29
Figura 29. Gota com bordas interiores e exteriores conectadas. ....	30
Figura 30. Imagem binária das gotas após filtro de detecção e fechamento. ....	30
Figura 31 – Resultados obtidos da validação do método.....	42

## SUMÁRIO

1. Introdução .....	1
2. Fundamentação Teórica.....	3
2.1 Tecnologia de Aplicação .....	3
2.2 Visão Computacional .....	3
2.3 Processamento de Imagens Digitais .....	3
2.4 Suavização de Imagem .....	4
2.5 Detecção de Bordas .....	4
2.6 Aprendizado de Máquina.....	6
2.7 Trabalhos Relacionados .....	6
3. Material e Métodos.....	10
3.1 Etapas Realizadas .....	10
3.2 Aquisição .....	11
3.3 Seleção da Resolução .....	11
3.4 Seleção Da Abertura da Lente.....	12
3.5 Seleção do Tempo de Exposição.....	14
3.6 Pré-Processamento (Suavização de Imagem) .....	14
3.7 Segmentação (Detecção de Bordas).....	16
3.8 Teste com Filtros de Detecção de Bordas.....	17
3.8.1 Prewitt.....	17
3.8.2 Sobel.....	20
3.8.3 Laplaciano.....	22
3.8.4 <i>Canny Edge</i> .....	23
3.9 Segmentação (Morfologia Matemática) .....	28
3.10 Segmentação por agrupamento (Aprendizado de Máquina).....	30
3.11 Extração de Características .....	31
3.12 Interpretação.....	32
4. Validação .....	33
5. Considerações Finais .....	34
6. Referências.....	35

# 1. INTRODUÇÃO

A tecnologia da informação pode ser aplicada em diversas áreas, auxiliando em problemas cotidianos e problemas específicos. Dentro da tecnologia da informação a área Visão Computacional tem obtido destaque quando tem-se a necessidade de obter dados de entrada que não podem ser descritos por palavras. Esta área tem como ciência estudar e implementar tecnologias que fazem máquinas terem uma visão biológica artificial, e podendo utilizar processamento de imagens em conjunto com inteligência artificial como cognição para extrair e processar dados em imagens.

O advento da chamada agricultura de precisão vem cada vez mais contribuindo para a aquisição de dados com maior rapidez e qualidade na informação, com o processamento dos mesmos via métodos computacionais. No estágio atual da agricultura, a eletrônica embarcada e o processamento de dados via softwares é uma constante. Hoje encontra-se no mercado máquinas com sofisticados sistemas de posicionamento global (GPS) delimitando áreas, gerenciando tarefas, identificando culturas por meio de índices de vegetação normatizados, bem como atuando no controle localizado de plantas daninhas, insetos, doenças e pragas.

A maioria dos produtos fitossanitários utilizados na agricultura para o controle de pragas, doenças e plantas daninhas é aplicada via pulverização, por meio de sistemas compostos por pontas hidráulicas, tanque, bomba, tubulações e mecanismos de acionamento. Sabe-se que o tamanho de gotas é influenciado pelo produto (calda), pressão de trabalho e tipos de pontas de pulverização que geram uma alta variabilidade no tamanho das gotas e velocidade.

Entre as diferentes técnicas de aplicação de agrotóxicos, as que se baseiam na pulverização hidráulica são as mais difundidas, graças à flexibilidade que oferecem em diferentes situações, tendo como objetivo colocar a quantidade certa de ingrediente ativo no alvo, com a máxima eficiência e de maneira econômica, afetando o mínimo possível o ambiente (TEIXEIRA, 1997; MATTHEWS, 2002).

Oliveira (2011) afirma que a alta demanda com agrotóxicos pelo sistema de produção agrícola aliada ao possível uso inadequado destes produtos torna a deriva um dos maiores problemas potenciais da agricultura atual, ocasionando riscos à saúde humana, impactos negativos ao meio ambiente, além de aumentar os custos de produção e reduzir os lucros dos produtores.

Shapiro e Stockman (2000), afirmaram que cientistas e escritores de ficção científica ficaram fascinados com a possibilidade de construir máquinas inteligentes, e a capacidade de

entender o mundo visual é um pré-requisito que alguns exigiriam de tal máquina. Grande parte do cérebro humano é dedicada à visão. Os humanos resolvem muitos problemas visuais sem esforço, mas a maioria tem um pouco de compreensão analítica da cognição visual como um processo. Allan Turing, um dos pais do moderno computador digital e do campo de inteligência artificial, acreditava que um computador digital alcançaria a inteligência e a capacidade de entender cenas. Tais metas elevadas provaram ser difíceis de alcançar e a riqueza da imaginação humana ainda não é igualada por nossa engenharia.

Considerando que nos catálogos fornecidos por fabricantes de pontas de pulverização a representação do tamanho das gotas geradas é realizada apenas em classificação por cores sem a informação numérica do tamanho. Outro fator a destacar é que os aparelhos presentes no mercado que fazem esta determinação possuem preços muito elevados. Neste contexto, objetivou-se com este trabalho, à um menor custo, aplicar diferentes técnicas de processamento de imagens para identificar e determinar tamanhos de gotas em imagens obtidas por meio de câmera *ultra slow-motion*.

## **2. FUNDAMENTAÇÃO TEÓRICA**

### **2.1 TECNOLOGIA DE APLICAÇÃO**

Na literatura Tecnologia de Aplicação é descrita como a ciência que investiga técnicas para medir a qualidade de aplicação dos produtos fitossanitários. O seu principal objetivo é maximizar a eficiência da aplicação de maneira econômica, depositando a quantidade certa de produto no alvo, minimizando a contaminação ao meio ambiente. (MATTHEWS, 2002; NUYTTENS, 2007; JACTO, 1999; MATUO, 1998).

O tamanho das gotas está entre as principais vertentes da qualidade da aplicação, conforme afirmado por Lefebvre (1989) e Márquez (1997), as gotas que possuem diâmetro menor que 100  $\mu\text{m}$  são facilmente arrastadas pelo vento e as gotas com diâmetro maior 800  $\mu\text{m}$  escorrem pela superfícies das folhas das plantas.

### **2.2 VISÃO COMPUTACIONAL**

A utilização de recursos computacionais para a simulação da visão biológica em máquinas é chamada visão computacional (Lemley et al., 2017, Wang et al., 2015). Obter informações de dados que não podem ser descritas por palavras é uma tarefa difícil, por este motivo a Visão Computacional tem obtido destaque em diversas áreas.

Diariamente a visão é utilizada para observar objetos e ambientes para obter informações relevantes. A visão é uma habilidade básica e comum para pessoas normais, porém os princípios e o processo do sistema visual são complexos afirmam Rautaray e Agarwal (2015).

Learned-Miller (2011), afirma que a maioria dos cientistas de visão computacional concordariam que ver significa mais que o processo de gravar luz em um formato para reprodução. Visão é interpretar imagens que auxiliam em tomadas de decisões, os processos de reconhecimento, interpretação, aprendizado possuem influência em tomadas de decisões.

### **2.3 PROCESSAMENTO DE IMAGENS DIGITAIS**

O processamento de imagens envolve um conjunto de tarefas para se alcançar um resultado, as quais são: aquisição, pré processamento, segmentação, extração de atributos e interpretação ou classificação.

De acordo com Queiroz (2001), o processamento de imagens envolve um conjunto de tarefas. A primeira tarefa é a aquisição da imagem, neste momento a imagem é capturada

através da luz refletida nos objetos e por um sistema de aquisição. Após a aquisição, realiza-se o pré-processamento, o qual deve remover ruídos introduzidos pelos sensores da câmera, e correções de distorções na imagem. Para se identificar objetos, é necessário extrair atributos da imagem, como bordas, textura e vizinhanças. E então, separamos estes objetos do plano de fundo (*background*) na etapa de segmentação. A classificação é considerada uma das tarefas de mais alto nível, pois esta etapa tem o objetivo de reconhecer, verificar ou inferir a identidade dos objetos a partir de suas características obtidas nos passos anteriores.

## 2.4 SUAVIZAÇÃO DE IMAGEM

A suavização de uma imagem tem como objetivo melhorar uma imagem para que possa ser processada, suavizando detalhes de uma imagem para remover ruídos presentes nas imagens, estes ruídos podem ser causados por reflexos de luz, poeira, pouca iluminação, entre outros, em processamento de imagens é realizado no momento de pré-processamento.

Será abordado o filtro Gaussiano, que de acordo com Jesus e Costa (2015), consideram o filtro Gaussiano como um filtro passa baixa, dizem também, que é utilizado para borrar ou desfocar uma imagem, com o objetivo de reduzir os ruídos presentes. Esta operação resulta em uma imagem desfocada, ou seja com os detalhes suaves, como se a imagem fosse vista através de uma lente fora de foco. O filtro de suavização Gaussiana é amplamente utilizada na etapa de pré-processamento de uma imagem para realçar a estrutura da imagem em diferentes escalas. A aplicação do filtro Gaussiano é realizado convoluindo a imagem com uma função Gaussiana,

## 2.5 DETECÇÃO DE BORDAS

Em processamento de imagens digitais detectores de bordas são amplamente utilizados pois tem a finalidade de realçar detalhes de objetos contidos na imagem, na literatura existem muitos detectores de bordas, tais como: *Canny Edge*, detectores de *Sobel*, *Laplaciano*, *Prewitt*, entre outros.

Canny (1986) afirma que os detectores de algum tipo de borda, particularmente os detectores de bordas do tipo degrau tem sido uma parte essencial para muitos sistemas de visão computacional. O processo de detectar bordas simplifica a análise das imagens, pois reduz a quantidade de dados a serem processados, ao mesmo tempo que preserva as informações da estrutura da imagem.

Segundo Radhika (2013) o operador de Sobel possui uma desvantagem com a relação sinal-ruído, a degradação das bordas aumenta conforme o ruído, o que resulta em uma perda

de precisão. Nenhum operador (Sobel, Prewitt e Roberts) consegue detectar precisamente as bordas descartando ruídos, porém o detector de bordas de Canny possui uma boa relação de sinal-ruído. O Canny Edge tem apresentado bons resultados, e pode ser utilizado como algoritmo de detecção de bordas em várias situações.

Saini et al. (2010) concordam que o detector de bordas criado por John Canny é amplamente considerado o algoritmo detector de bordas padrão na indústria, e ainda supera muitos algoritmos que foram desenvolvidos.

Canny (1986) apresenta três critérios para uma boa detecção de bordas:

1. Boa Relação Sinal/Ruído: um detector com uma boa relação sinal/ruído deve selecionar apenas bordas reais, não detectando então ruídos presentes na imagem.
2. Localização (L): as bordas detectadas devem estar o mais próximo possível das bordas reais.
3. Resposta Única: não deve ser detectado bordas duplicadas onde houver apenas uma borda real.

Atingindo estes critérios pode-se obter uma boa classificação das bordas, a relação sinal/ruído pode ser atingida suavizando os ruídos na imagem para que não sejam detectados, o critério de localização é atingido quando a borda detectada está próxima do centro da borda real e o critério de resposta única quando se detecta apenas um *pixel* de borda por borda real, as bordas que apresentarem mais de um *pixel* detectado, deverá marcar apenas um como borda real e descartar os outros *pixels* de borda.

O processo de detecção é realizado em quatro etapas, explicadas por (Canny, 1986) como:

1. Suavizando a imagem com uma máscara Gaussiana para reduzir o efeito ruidoso da imagem.
2. Calculando a direção das bordas da imagem, as bordas podem ter várias direções, para isto são utilizadas quatro máscaras detectoras (horizontal, vertical e diagonais com 45° e 135°).
3. Para cada pixel da imagem convoluída armazena-se a maior intensidade e o sentido do gradiente.
4. Os gradientes de maior intensidade tem uma probabilidade maior de ser uma borda real, não existe então um valor de intensidade pré-definido para se determinar uma borda, para isto utiliza-se então uma técnica chamada de “Limiarização com Histerese” que dado dois valores de limiarização, um superior e um inferior, as bordas que estiverem dentro destes valores podem ser seguidas e detectadas.

Maini e Aggarwal (2009) provam que o custo computacional do detector de bordas do Canny é mais elevado comparado aos detectores de bordas Sobel, Prewitt e de Robert. Porém, o algoritmo de detecção de bordas de Canny tem melhor desempenho que todos estes outros detectores em quase todos cenários.

## **2.6 APRENDIZADO DE MÁQUINA**

Segundo Michalski (2013), aprendizado de máquina define-se em programar computadores para otimizar um critério de desempenho usando dados de exemplo, ou experiência passada. Onde precisa-se aprender nos casos em que não se pode escrever um programa de computador para resolver um determinado problema diretamente, por precisar de exemplo de dados ou experiência. Um caso em que o aprendizado de máquina é necessário é quando a perícia humana não existe, ou quando os seres humanos são incapazes de explicar seus conhecimentos.

O algoritmo de clusterização espacial baseada em densidade de aplicações com ruído (DBSCAN) em inglês “Density Based Spatial Clustering Algorithm for Applications with Noise” proposto por Ester et al. (1996) é considerado um algoritmo de agrupamento não-supervisionado pois permite a identificação de grupos de diversos formatos e tamanhos sem uma informação prévia dos dados, permite também separar dados considerados como ruídos dos grupos identificados.

Ester et al. (1996) cita que o DBSCAN pode ser aplicado à qualquer base de dados que possibilite o cálculo de distâncias entre os dados. Para calcular a distância entre os pontos quaisquer função de distância pode ser utilizada, e deve-se selecionar uma de acordo com o contexto da base de dados.

Para agrupar os dados o DBSCAN utiliza dois parâmetros, raio e número mínimo de pontos, então dentro do raio de cada ponto dos dados, deve-se ter um número mínimo de vizinhos para que se possa definir um grupo (Ester et al. 1996).

## **2.7 TRABALHOS RELACIONADOS**

Alguns trabalhos foram desenvolvidos no intuito de identificação de gotas utilizando métodos computacionais, sendo que Bachalo e Houser (1984) foram os precursores desta técnica analisando espectro de gotas por meio do ângulo da dispersão do feixe de luz causada pelas gotas. Esta técnica permite a análise do tamanho de gotas no formato esférico limitando-se em analisar gotas em outros formatos.

Instrumentos de medição de tamanho de gotículas de difração a laser, como o Malvern 2600 ou o Sympatec Helios Vario e o Doppler de Fase analisadores (PDPA) têm sido o instrumento mais comum usado para medir o tamanho das gotas e é relativamente rápido e fácil de usar.

Cunha, Teixeira e Fernandes (2007) estudaram a determinação do espectro de gotas de pulverização utilizando método de análise baseada em difração à laser por meio de um equipamento Spraytec (Malvern Spraytec Real Time Droplet Sizing System, Malvern Instruments, Malvern, England). O equipamento dispõe de unidade óptica que detecta o padrão de difração da luz ao passar por um conjunto de partículas (gotas), apresentando capacidade de mensuração com diâmetro entre 0,5 e 1.000  $\mu\text{m}$ .

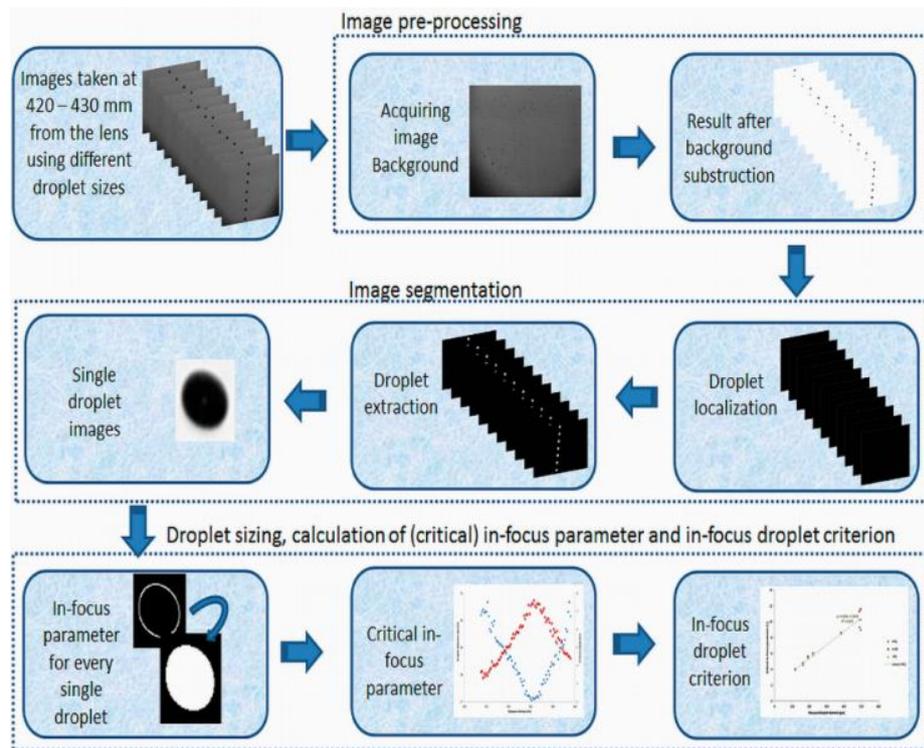
Vulgarakis et al. (2016), afirmaram que o desenvolvimento de tecnologias computacionais e sistemas a lasers permitem a criação de novos métodos para determinação das dimensões das gotas em pontas de pulverização. Citam também que técnicas de dispersão a luz não-quantitativa e óptica estão sendo desenvolvidas para o mesmo fim, como a caracterização da pulverização na definição não-intrusiva de jatos.

França et al. (2018), utilizaram um analisador de gotas em tempo real (VisiSize Portable P15) da Oxford Lasers, Imaging Division, Oxford, U.K., que se baseia em alta resolução de análise de imagem. O sistema analisa o espectro de gotas usando uma técnica chamada Análise de Imagem de Partículas / Gotas (PDIA) que consiste na identificação da gota ao passar por um espectro de luz.

(Carvalho et al., 2017), adotaram o sistema descrito acima, por conseguir determinar gotas com diâmetro maiores que 5  $\mu\text{m}$  e também a velocidade das gotas simultaneamente. Neste sistema pode-se definir a determinação do espectro de gotas pelo tempo de coleta ou pelo quantitativo de gotas por repetição.

Vulgarakis et al. (2016) apresentaram o princípio de uma técnica baseada no processamento de imagens para medir o tamanho e a velocidade das gotas de pontas de pulverização usando um sistema de aquisição de imagem desenvolvido, conforme a Figura 1. A técnica foi desenvolvida utilizando software Matlab para processar as imagens.

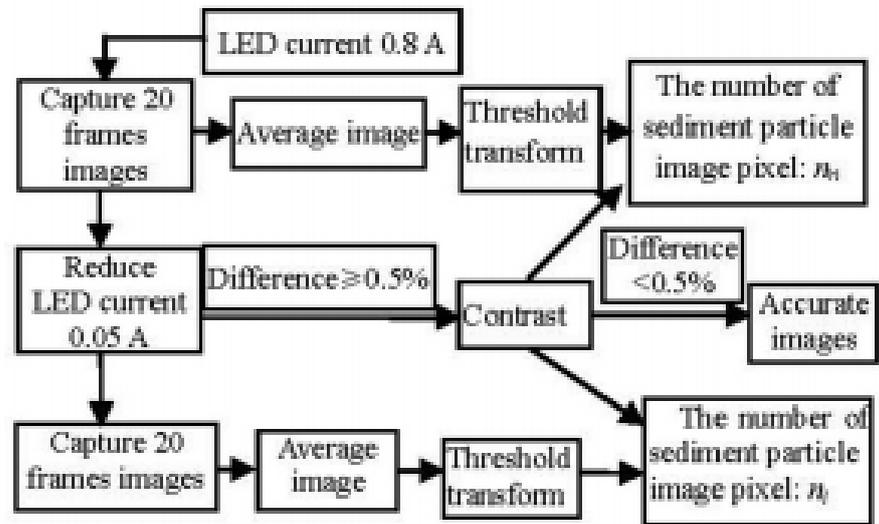
Figura 1 – Etapas de processamento de imagens (Vulgarakis et al., 2016).



1. Aquisição: utilizaram um protocolo para definição de distâncias e configuração dos objetos para captura (câmera, luz e gotas) e resolução de imagem de 1280x1024 *pixels*.
2. Pré-processamento: para resolver o problema de não ter um fundo homogêneo devido à padrões de luz e manchas na lente, removeram o plano de fundo da imagem utilizando 70 imagens para reconstrução de uma imagem apenas do plano de fundo, para então realizar uma normalização do plano de fundo nas imagens das gotas.
3. Segmentação de Imagem: identificando e recortando sub-imagens das gotas encontradas utilizando filtros de processamento de imagem.
4. Interpretação: aplicando os critérios *in-focus* (identificar as gotas) e *critical in-focus* (identificar as gotas focadas) em cada gota encontrada para definir se a gota é uma gota focada ou uma gota desfocada.

Chen et al. (2013) estudaram métodos para determinação do diâmetro de partículas sólidas utilizando iluminação por LED, câmera fotográfica com imagem preto e branco utilizando resolução de 1628x1236 e uma mesa vibradora para seleção em quatro categorias de tamanhos de partículas. A Figura 2 mostra o fluxograma do método estudado.

Figura 2 – Fluxograma estudado por Chen et al. (2013)



### 3. Material e Métodos

Este trabalho foi desenvolvido no laboratório do Núcleo de Investigação em Tecnologia de Aplicação de Agroquímicos (NITEC) da Universidade Estadual do Norte do Paraná (UENP), Campus Luiz Meneghel, Bandeirantes/PR.

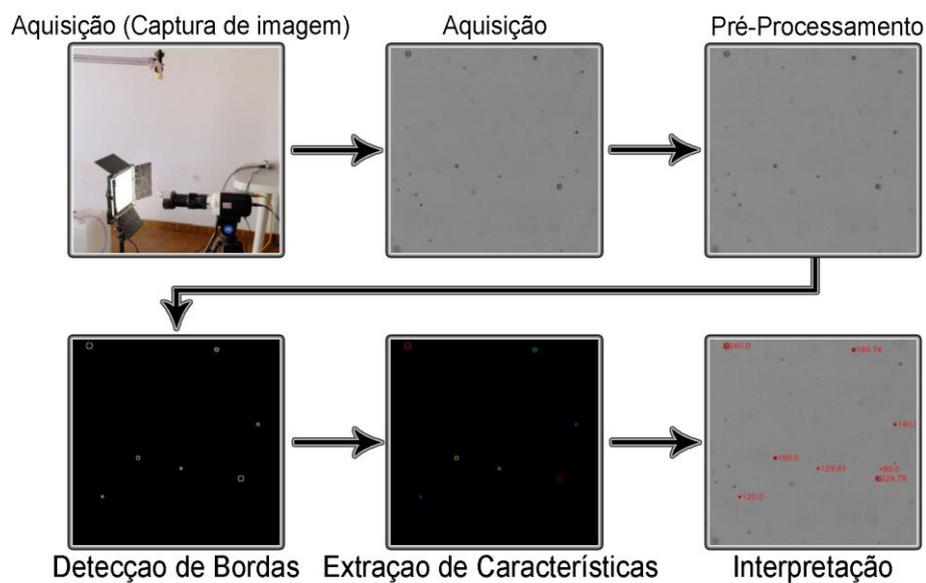
Os protótipos foram desenvolvidos utilizando a linguagem de programação *Python* com o módulo de processamento de imagens *OpenCV* e um protocolo pré-definido para configurações dos componentes utilizados para aquisição de imagens (câmera, luz e geração de gotas).

O processo de validação foi realizada comparando resultados do método proposto com os resultados do *ISCapture 2.2.1 software* criado pela empresa *Scienon Technology Co. Ltd.*, este *software* disponibiliza ferramentas de processamento de imagens para determinação manual de tamanhos de objetos em imagens de diferentes escalas. As imagens podem ser capturadas tanto utilizando uma câmera própria do equipamento quanto imagens externas.

#### 3.1 ETAPAS REALIZADAS

As etapas padrões do processamento de imagem adotadas neste trabalho foram: Aquisição, Pré-processamento, Segmentação, Extração de Características e Interpretação.

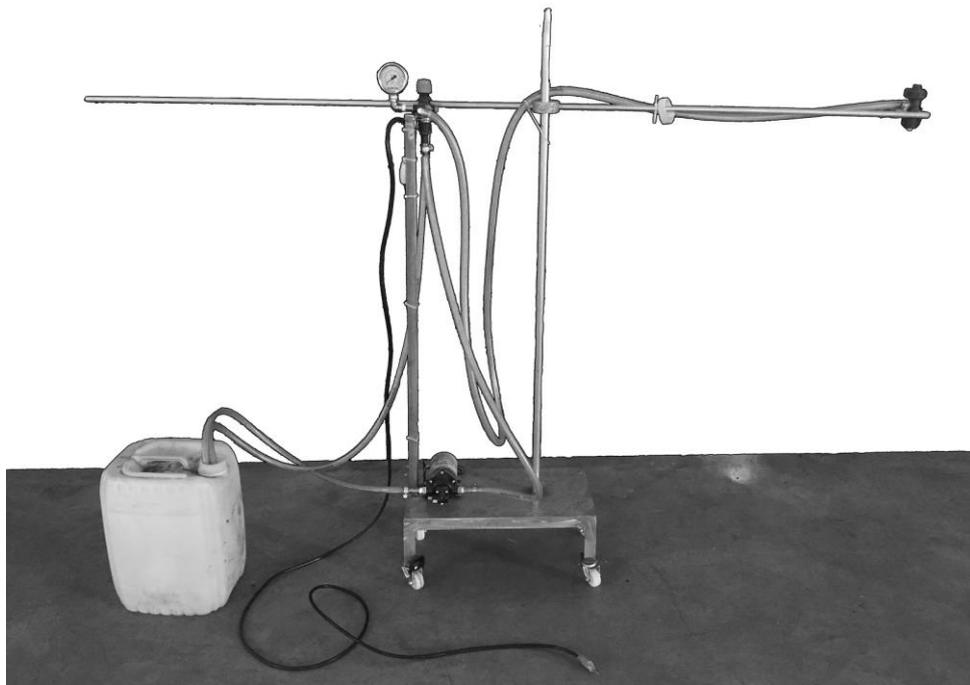
Figura 3. Resumo geral das etapas do processamento de imagens.



### 3.2 AQUISIÇÃO

A primeira etapa no processamento de imagens é a aquisição, nesta etapa será realizada a captura das imagens a serem processadas. A captura das imagens de gotas de pulverização é realizada por meio de um sistema de pulverização desenvolvido especificamente para este projeto. O sistema constitui de um depósito, bomba de sucção, circuitos hidráulicos, comando, manômetro, filtros e pontas de pulverização (Figura 1).

Figura 4 - Sistema simulador de pulverização.



Fonte: Acervo do autor.

Para aquisição das imagens foi desenvolvido em laboratório (NITEC) um protocolo, neste protocolo estão definidas as distâncias dos componentes utilizados para aquisição das imagens, as distâncias dos componentes definem exatamente a quantidade de luz e contraste obtido na imagem. Neste trabalho não será discutido o protocolo pois está em processo de patenteamento, porém são apresentados as seguintes configurações de imagem: resolução, abertura da lente e tempo de exposição.

### 3.3 SELEÇÃO DA RESOLUÇÃO

Deverá ser selecionada a resolução da imagem que será utilizada para capturar as gotas, para isto deverá ser testado diferentes resoluções. Sabendo que quanto maior a resolução, maior o custo de processamento, ao mesmo tempo quanto menor a resolução,

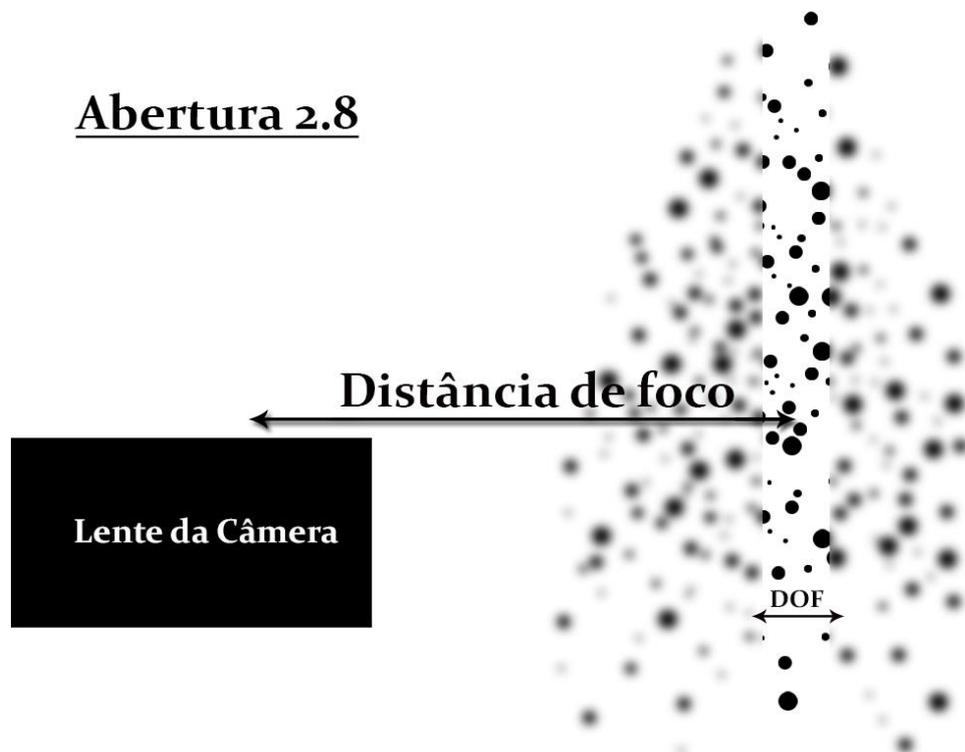
menor a qualidade dos detalhes da imagem. A resolução selecionada deverá manter um bom equilíbrio entre custo de processamento e qualidade dos detalhes da imagem.

A câmera utilizada captura imagens e grava em sua memória interna, após o término da captura, transfere as imagens para o computador por um cabo de rede. O armazenamento interno da câmera é limitado, então quanto maior a resolução, maior o peso do vídeo, e menor a quantidade de *frames* por segundo que podemos capturar.

### 3.4 SELEÇÃO DA ABERTURA DA LENTE

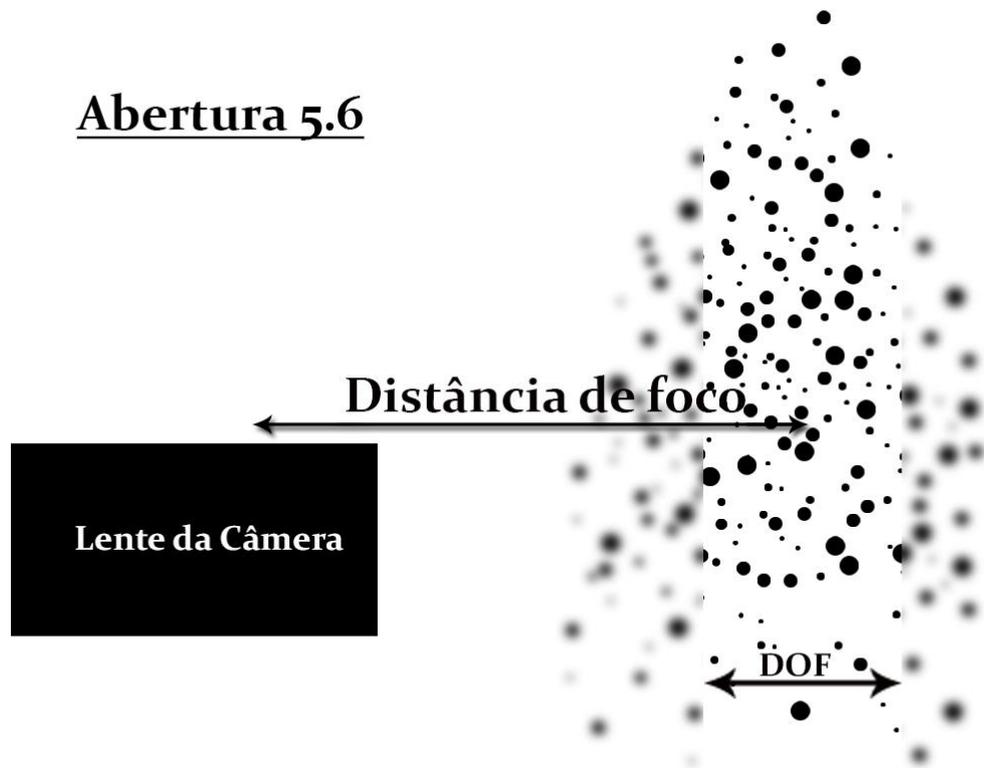
A abertura de lente determinará a profundidade de campo popularmente conhecida na literatura como DOF (*depth-of-field*), para isso testes foram realizados utilizando aberturas de 2.8 e 5.6. Quanto maior a abertura, maior o DOF, a seguir temos uma ilustração exemplo do aumento do DOF com o aumento da abertura de lente.

Figura 5 - Ilustração de profundidade de campo de abertura 2.8.



Fonte: Acervo do autor.

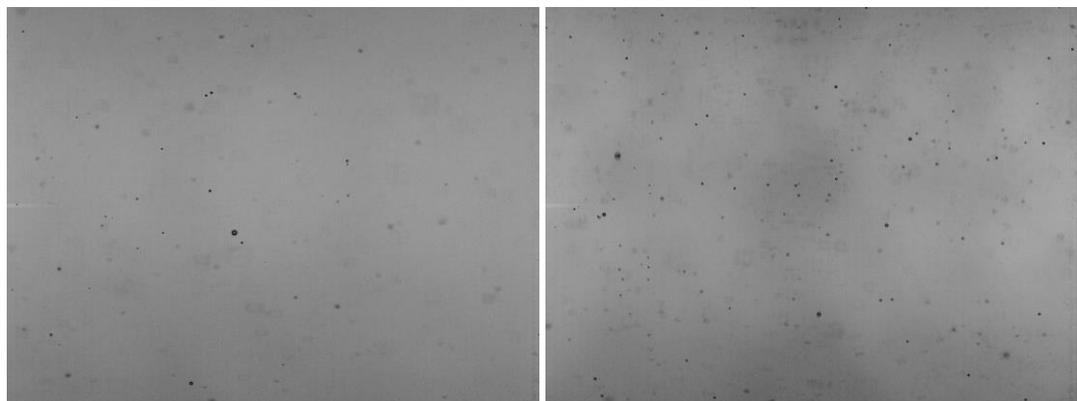
Figura 6 - Ilustração de profundidade de campo de abertura 5.6.



Fonte: Acervo do autor.

A profundidade de campo determina a região em foco do jato, com regiões maiores temos maior quantidade de gotas, e com regiões menores temos menor quantidade de gotas. A escala dos objetos na imagem é afetado por a distância entre câmera e objeto, não é possível obter a distância exata entre gota e lente, então é preferido utilizar uma região de foco menor pois a variação de escala entre as gotas é menor.

Figura 7 – Imagens de gotas com diferentes configurações de abertura de lente. (a) Abertura de lente 2.8. (b) Abertura de lente 5.6.



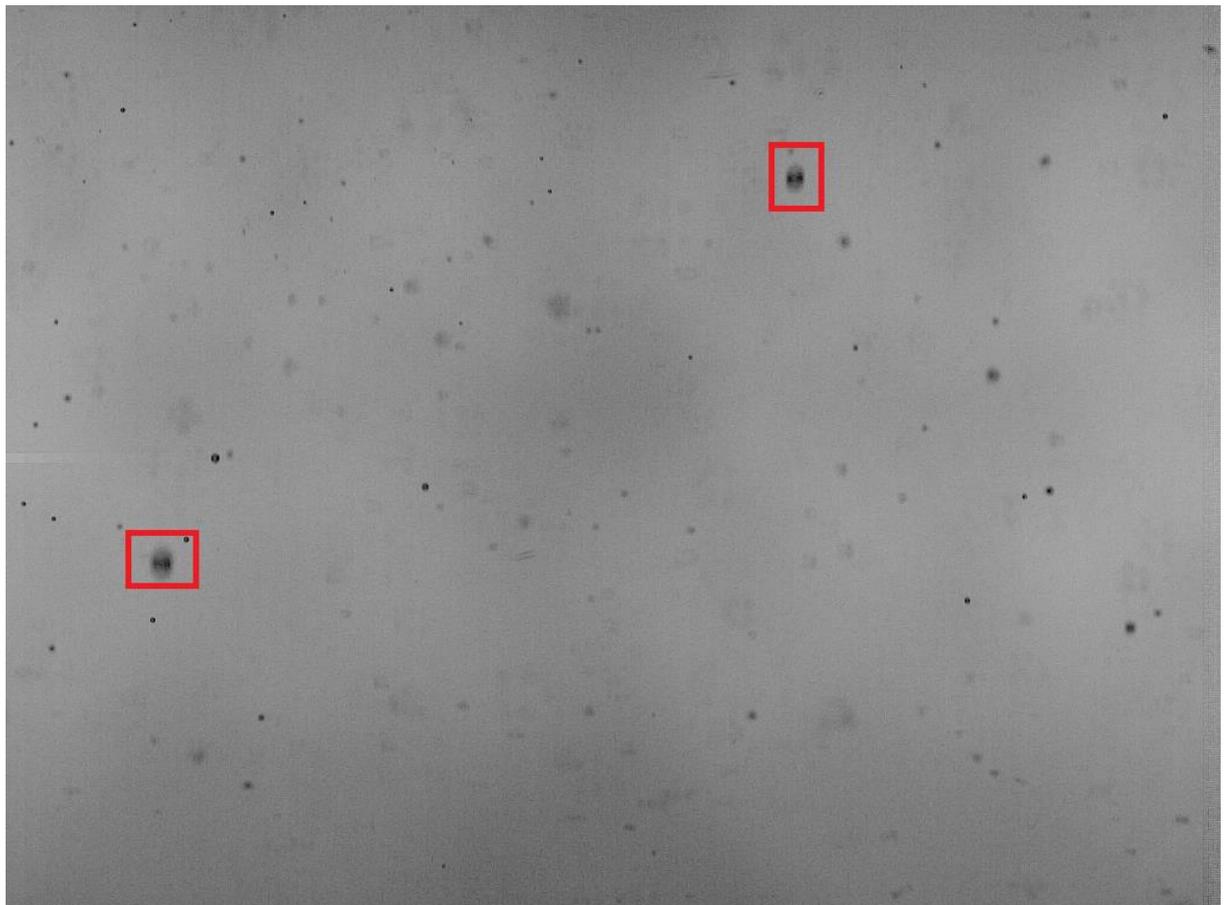
(a)

(b)

### 3.5 SELEÇÃO DO TEMPO DE EXPOSIÇÃO

O tempo de exposição determina o tempo em que o sensor da câmera fica exposto à luz, um intervalo de tempo muito curto pode fazer com que pouca luminosidade seja absorvida assim resultando em uma imagem com pouco contraste entre gotas e plano de fundo, e uma velocidade muito baixa pode fazer com que gotas muito velozes sejam capturadas borradas. Testes foram realizados com  $10\mu\text{s}$  e  $6\mu\text{s}$  para tempo de exposição.

Figura 8 – Imagem de gotas borradas com tempo de exposição de  $10\mu\text{s}$ .



O tempo de exposição de  $6\mu\text{s}$  permitiu capturar gotas sem borrar a imagem e mantendo uma boa relação de contraste entre gotas e plano de fundo.

### 3.6 PRÉ-PROCESSAMENTO (SUAVIZAÇÃO DE IMAGEM)

Esta etapa teve como objetivo reduzir ruídos na imagem e suavizar as bordas para que não seja detectada como uma borda serrilhada sem deformar as gotas na imagem.

A seleção deste filtro foi resultante de testes entre filtros de suavização de imagem (Desfoque Gaussiano e Desfoque Mediana).

Figura 9. Imagem das gotas sem filtro de suavização.

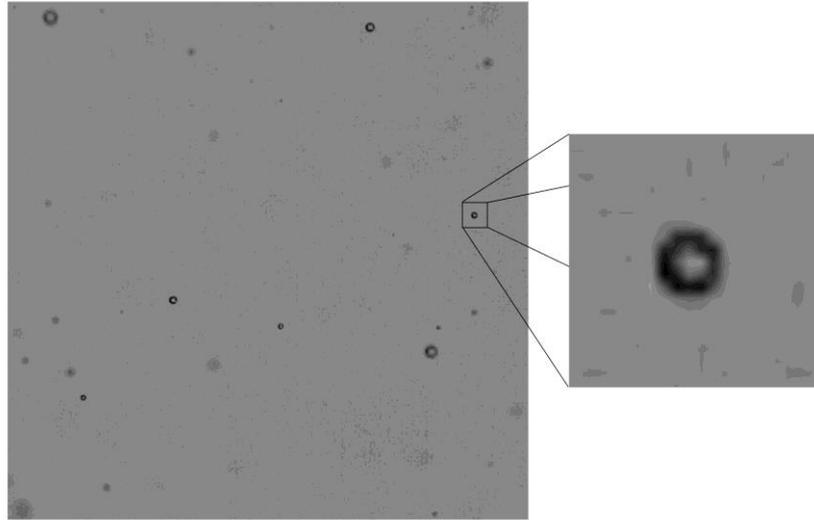


Figura 10. Imagem das gotas com filtro de suavização Mediana.

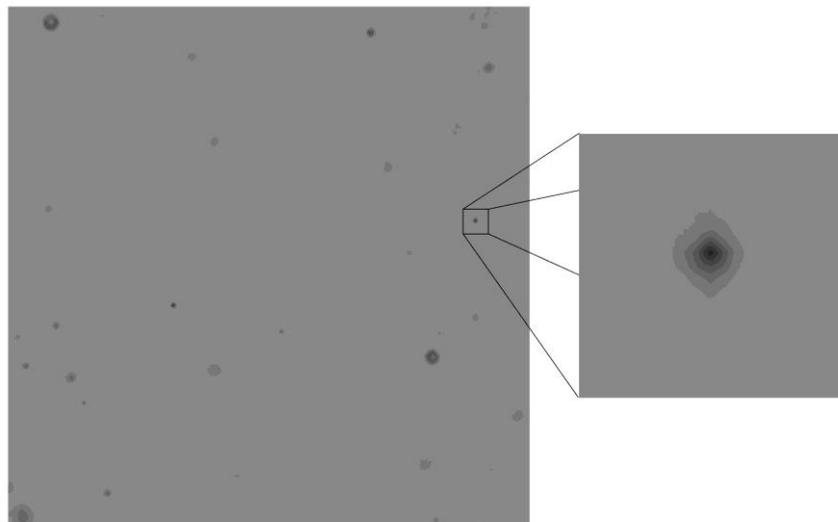
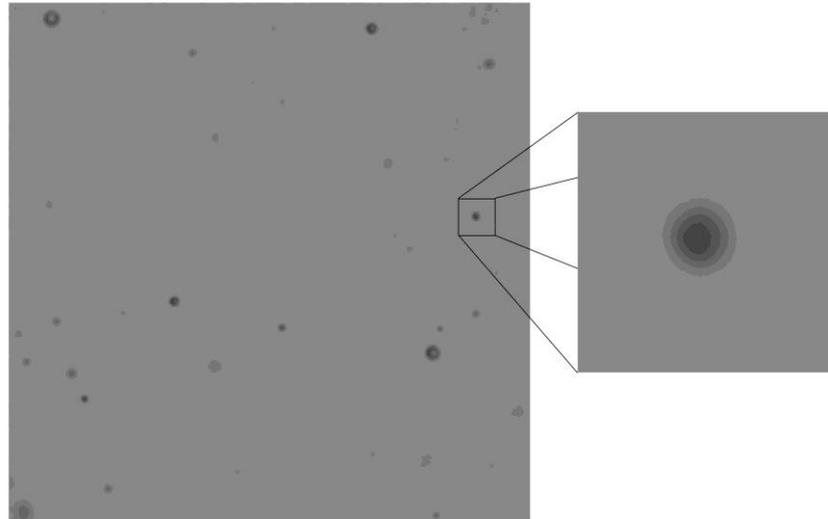


Figura 11. Imagem das gotas com filtro de suavização Gaussiana.

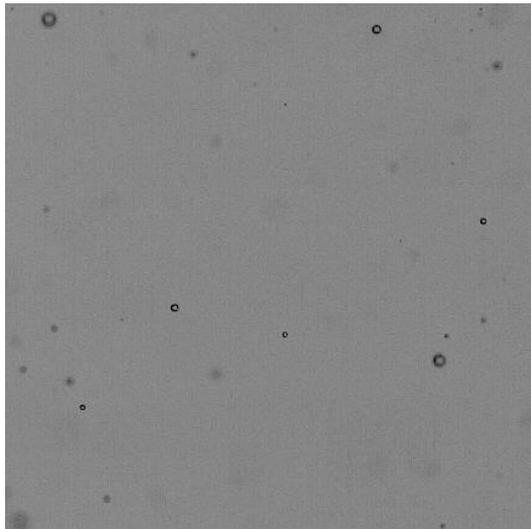


Comparando os filtros de suavização Mediana e Gaussiano pode-se observar que o filtro de suavização Mediana modificou o formato e tamanho das gotas na imagem, já o filtro de suavização Gaussiana não modificou o formato e tamanhos das gotas na imagem.

### 3.7 SEGMENTAÇÃO (DETECÇÃO DE BORDAS)

Nesta etapa, foi detectado as bordas dos objetos na imagem, para isto foi utilizado diferentes filtros de processamento de imagens a fim de detectar bordas de gotas focadas e descartar bordas de gotas desfocadas, pois as gotas focadas estão dentro de uma região de foco denominada de “Profundidade de Campo”, e esta distância permite calcular a escala dos objetos em imagem, as gotas desfocadas estão fora desta “Profundidade de Campo”, e não podem ser calculadas utilizando a mesma escala das gotas focadas. Os filtros testados foram: *Canny Edge*, *Prewitt*, *Sobel* e *Laplaciano*, todos testados com e sem filtros de suavização Gaussiano e com a mesma imagem (Figura 6).

Figura 12. Imagem em estado original utilizada para testes.



Nas Figuras 7, 8, 9, 10, 11, 12, 13, 14 e 15 serão demonstrados os diferentes filtros utilizados no trabalho.

### **3.8 TESTE COM FILTROS DE DETECÇÃO DE BORDAS**

#### **3.8.1 Prewitt**

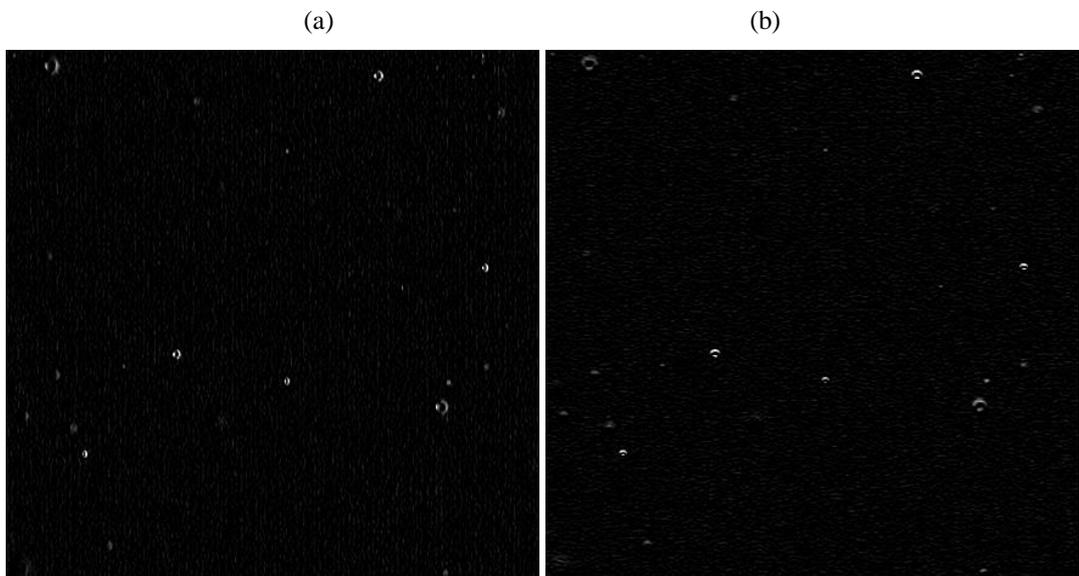
O filtro de detecção de bordas Prewitt Vertical foi aplicado utilizando *kernel* de 5x5:

$$\begin{bmatrix} -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix}$$

E Prewitt Horizontal com *kernel*:

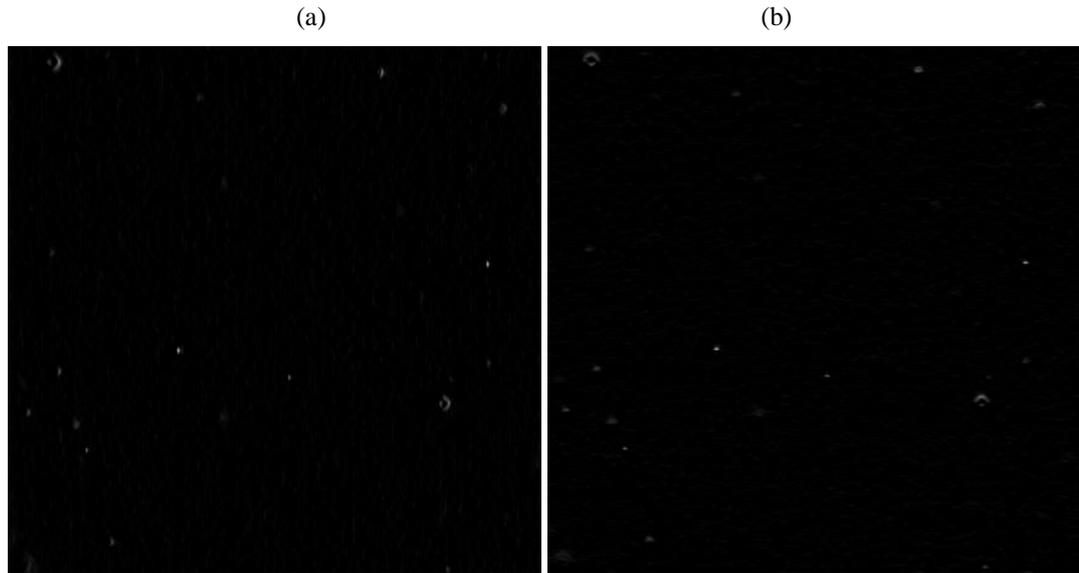
$$\begin{bmatrix} -2 & -2 & -2 & -2 & -2 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

Figura 13. Imagens utilizando Prewitt sem suavização. (a) Prewitt Horizontal (b) Prewitt Vertical



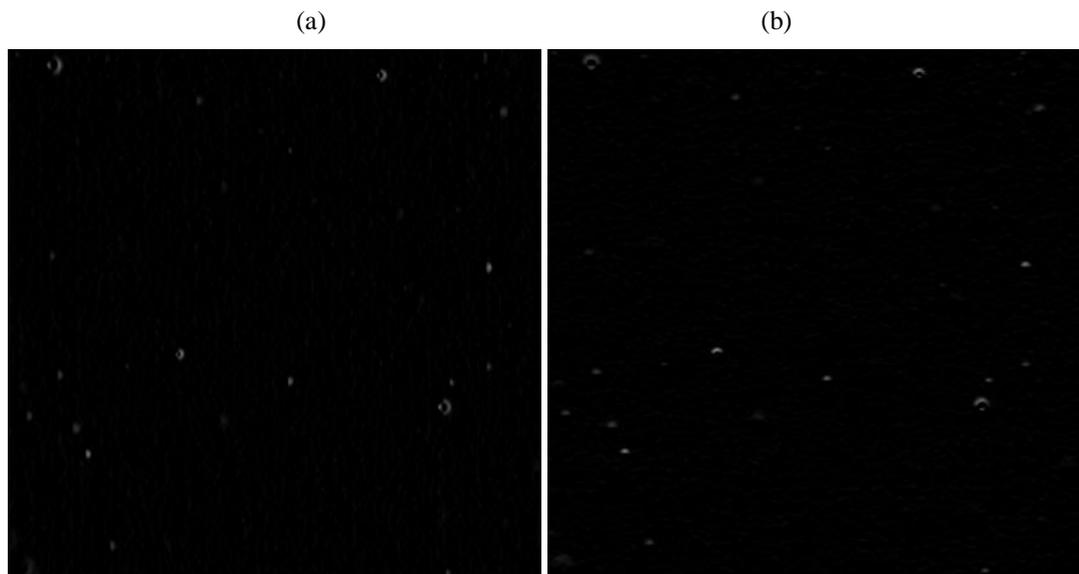
O filtro Prewitt tanto horizontal quanto vertical se demonstrou sensível aos ruídos da imagem, e também detectou parcialmente as bordas das gotas na imagem.

Figura 14. Imagens utilizando Prewitt com suavização Mediana. (a) Prewitt Horizontal (b) Prewitt Vertical



Combinando o filtro Prewitt com um filtro de suavização por Mediana fez com que diminuísse os ruídos da imagem, porém as bordas também diminuiram.

Figura 15. Imagens utilizando Prewitt com suavização Gaussiana. (a) Prewitt Horizontal (b) Prewitt Vertical



A combinação de Prewitt com filtro de suavização Gaussiana, resultou-se em uma redução de ruídos e uma detecção de mais bordas quando comparado ao Prewitt com filtro de Mediana, porém as bordas ainda apresentam ser de metade da gota.

O filtro de detecção de bordas Prewitt não demonstrou bons resultados para detecção das gotas nesta imagem pois mesmo suavizando a imagem, apresentou sensibilidade aos ruídos da imagem, selecionou gotas desfocadas e não torna possível definir o tamanho das gotas encontradas por meio das bordas detectadas, pois apresenta bordas parciais e largas.

### 3.8.2 Sobel

O filtro de detecção de bordas Sobel é baseado no filtro de detecção de bordas Prewitt, o que os difere são os pesos utilizados no *kernel*.

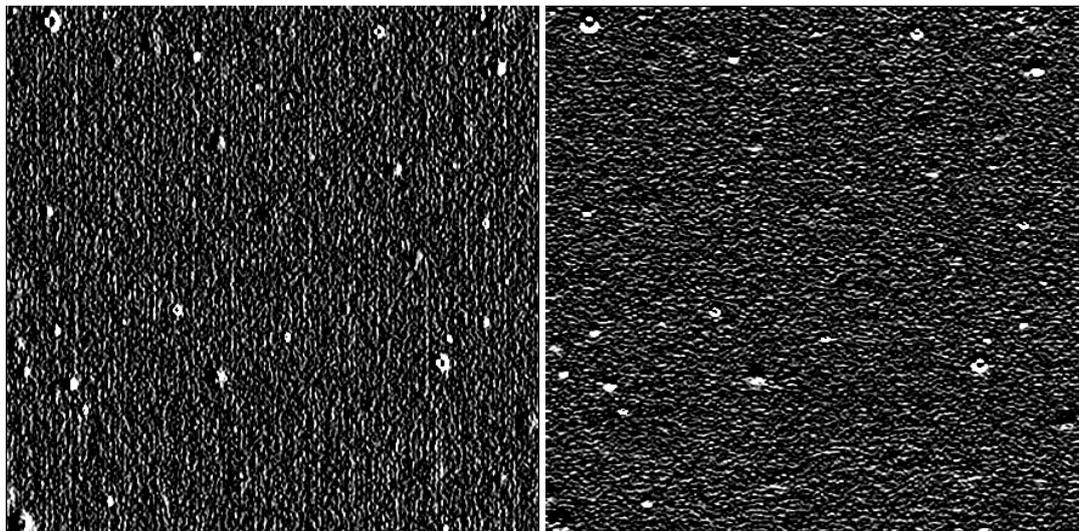
O filtro de detecção de bordas Sobel Vertical foi aplicado utilizando *kernel* de 5x5:

$$\begin{bmatrix} -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -4 & -2 & 0 & 2 & 4 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix}$$

E Sobel Horizontal com *kernel*:

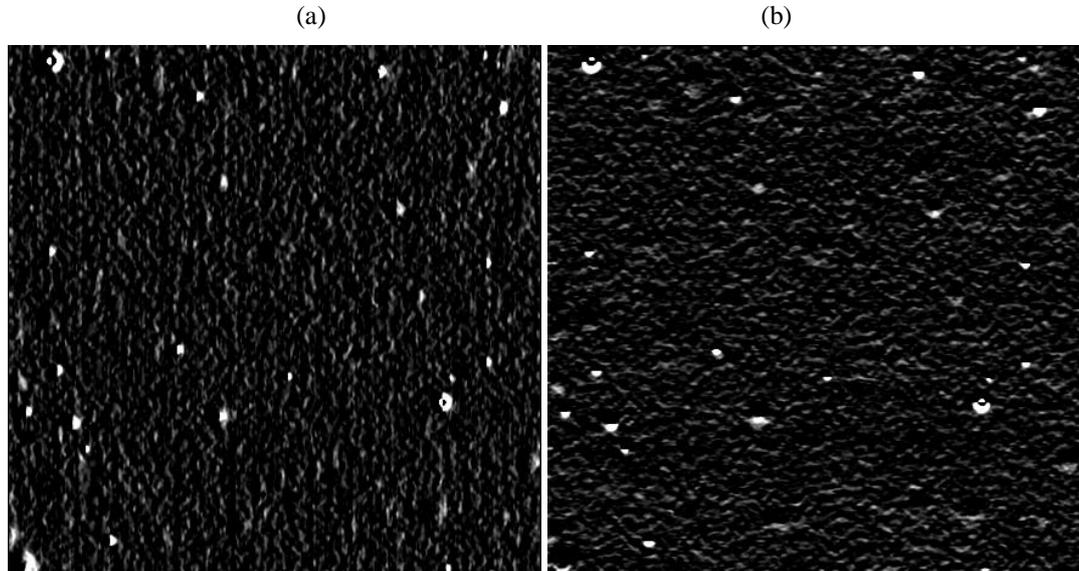
$$\begin{bmatrix} -2 & -2 & -4 & -2 & -2 \\ -1 & -1 & -2 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 1 & 1 \\ 2 & 2 & 4 & 2 & 2 \end{bmatrix}$$

Figura 16. Imagens utilizando Sobel sem suavização. (a) Filtro Sobel Horizontal (b) Filtro Sobel Vertical



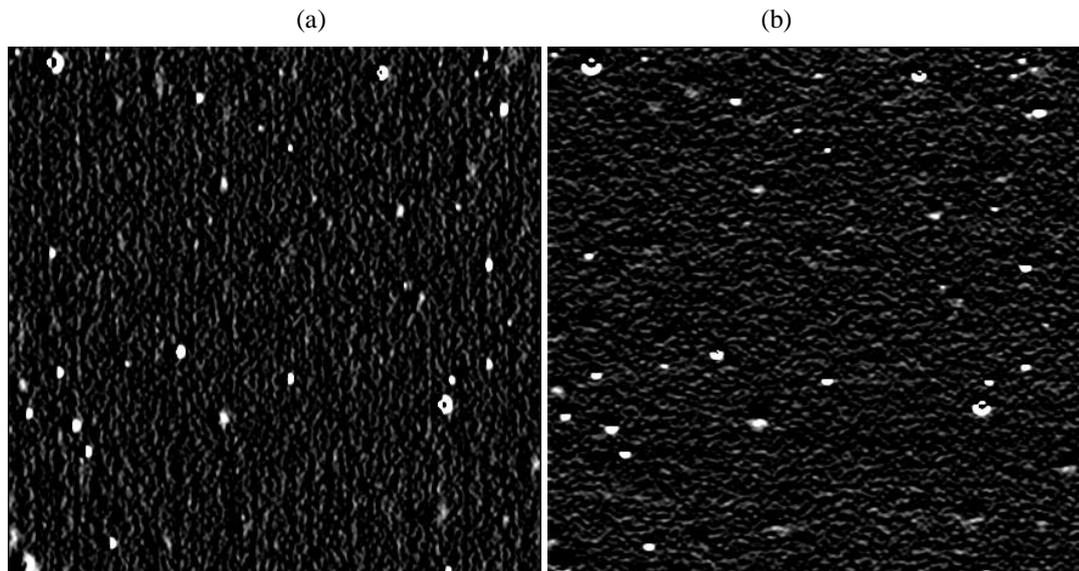
A imagem resultante do filtro Sobel apresenta uma grande quantidade de ruídos, tornando necessário uma suavização antes da aplicação do Sobel nesta imagem.

Figura 17. Imagens utilizando Sobel com suavização Mediana. (a) Filtro Sobel Vertical (b) Filtro Sobel Horizontal



A imagem resultante da combinação do filtro Sobel com um filtro de suavização Mediana resultou em uma imagem com uma grande quantidade de ruídos, porém com menos ruídos quando comparada com a imagem apenas com Sobel, demonstrando então que o filtro de suavização é necessário, sendo assim, testaremos o filtro Sobel com outro filtro de suavização.

Figura 18. Imagens utilizando Sobel com suavização Gaussiana. (a) Sobel Horizontal (b) Sobel Vertical



As imagens resultantes da combinação de Sobel e filtro de suavização Gaussiana resultou em uma imagem ruidosa, e com bordas parciais das gotas na imagem.

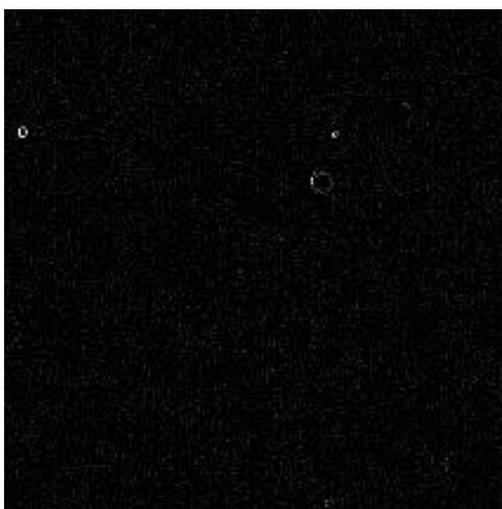
A detecção de bordas de Sobel apresentou sensibilidade aos ruídos da imagem, mesmo quando suavizada, as gotas desfocadas não foram descartadas e apresenta bordas de somente metade da gota.

O filtro de detecção de bordas Sobel não demonstrou bons resultados para detecção das gotas nesta imagem pois mesmo suavizando a imagem, apresentou sensibilidade aos ruídos da imagem, selecionou gotas desfocadas e não torna possível definir o tamanho das gotas encontradas por meio das bordas detectadas, pois apresenta bordas parciais e largas.

### 3.8.3 Laplaciano

O filtro Laplaciano foi aplicado utilizando um *kernel* 3x3:

Figura 19. Imagem utilizando Laplaciano sem filtro de suavização.



A imagem resultante do Laplaciano apresentou bordas completas das gotas na imagem, porém também detectou ruídos da imagem e gotas desfocadas. Sendo necessário a aplicação de um filtro de suavização.

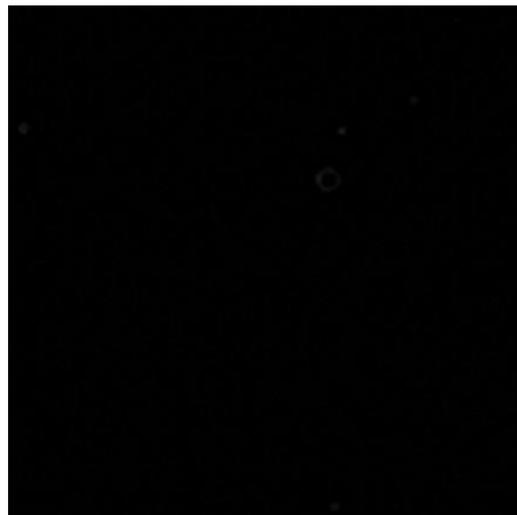
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Figura 20. Imagem com Laplaciano com filtro de suavização Mediana.



A aplicação do filtro de suavização reduziu ruídos na imagem, porém ainda apresentou ruídos que não deveriam ser detectados.

Figura 21. Imagem com Laplaciano utilizando suavização Gaussiana.



A aplicação de suavização Gaussiana antes do filtro Laplaciano apresentou ruídos na imagem, porém uma redução de ruídos maior que a suavização Mediana, entretanto também detectou bordas de gotas desfocadas.

#### 3.8.4 *Canny Edge*

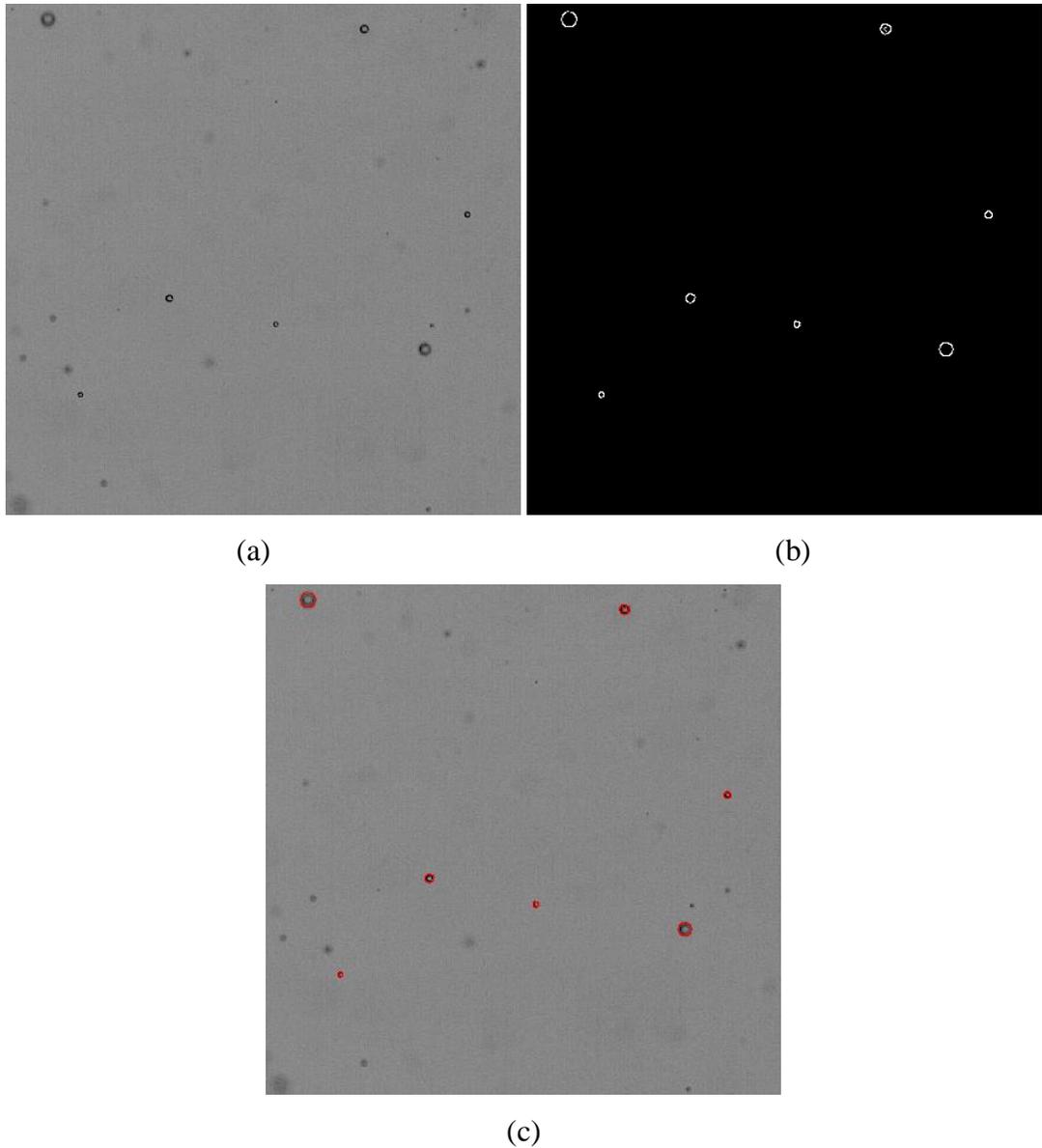
O filtro de detecção de bordas *Canny Edge* segue o método proposto na dissertação de mestrado do Massachusetts Institute of Technology - MIT (Canny, J., 1983).

O filtro *Canny Edge* tem como resultado uma imagem binária, onde as bordas detectadas aparecem em cor branca (255) e o restante da imagem aparece em preto (0).

A aplicação do filtro *Canny Edge* necessita de uma imagem previamente suavizada e dois parâmetros de limiarização, limiarização máxima e mínima. Cada borda da imagem para

ser detectada deve ter algum ponto acima do valor de limiarização máximo e um ponto abaixo do valor de limiarização mínimo, para ser considerada uma borda real.

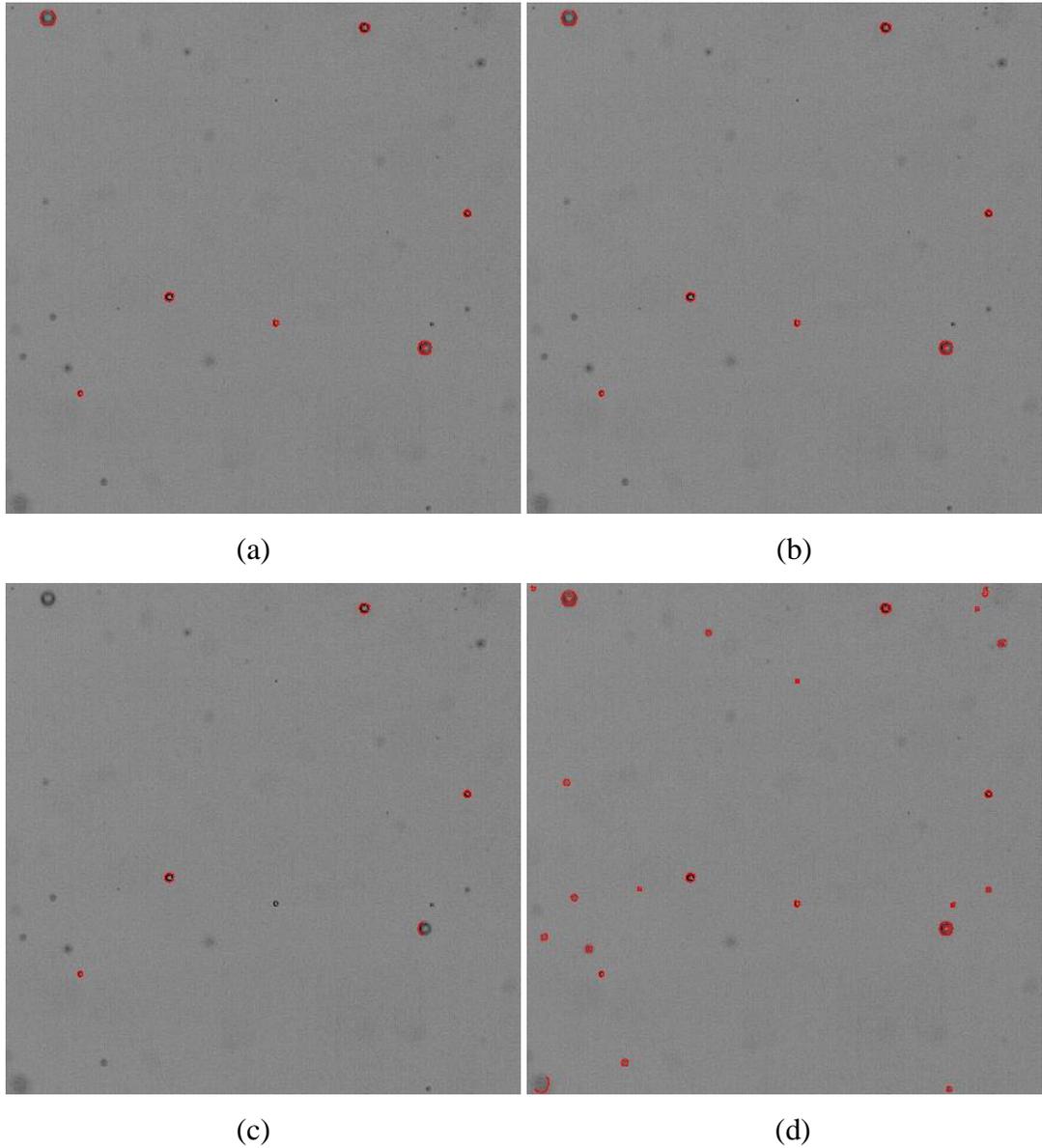
Figura 22. Imagem com *Canny Edge* utilizando suavização Gaussiana. (a) Imagem original. (b) Imagem binária resultante do *Canny Edge*. (c) Imagem original com bordas detectadas na imagem binária em vermelho.

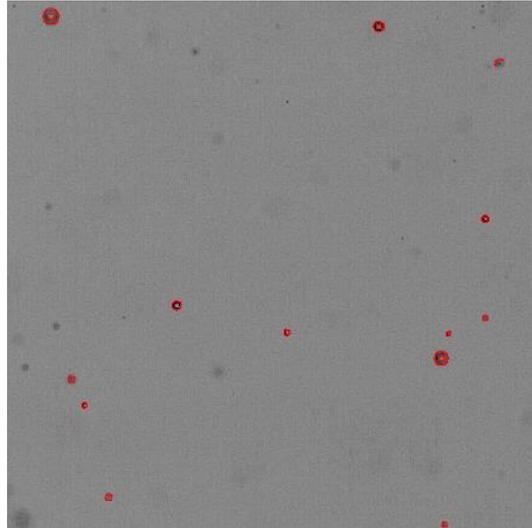


O filtro *Canny Edge* demonstrou uma ótima seleção das bordas das gotas na imagem, pois não detectou ruídos da imagem, detectou bordas ao redor da gota toda e não selecionou gotas desfocadas. Os valores utilizados para limiarização mínima e máxima foram encontrados a partir de uma técnica proposta por Rosebrock (2015).

O valor de *sigma* utilizado na técnica de Rosebrock foi 0,33, que determina valor 33% acima da mediana do tom de cinza para valor máximo de limiarização e valor 33% abaixo da mediana do tom de cinza para valor mínimo de limiarização, pois é o valor que demonstrou melhores resultados.

Os valores testados do *Canny Edge* antes da aplicação da técnica de Rosebrock foram: Figura 23. Testes com valores para *Canny Edge*. (a) Imagem aplicando *Canny Edge* com valores 0 e 50 para mínimo e máximo. (b) Imagem aplicando *Canny Edge* com valores 50 e 100 para mínimo e máximo. (c) Imagem aplicando *Canny Edge* com valores 100 e 150 para mínimo e máximo. (d) Imagem aplicando *Canny Edge* com valores 150 e 200 para mínimo e máximo. (e) Imagem aplicando *Canny Edge* com valores 200 e 250 para mínimo e máximo.

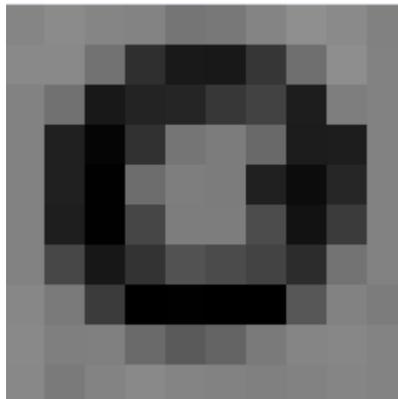




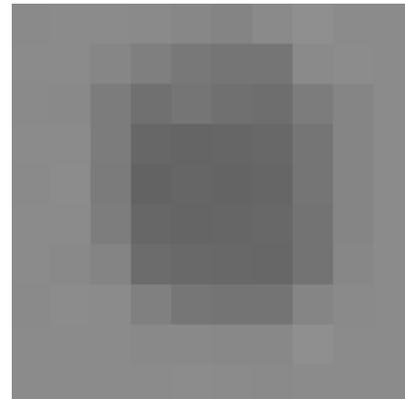
(e)

Os resultados do *Canny Edge* com diferentes valores mínimo e máximo de limiarização com intervalo de 50 apresentam que a limiarização realizada em tons mais escuros (próximos de zero) apresentam detecção de gotas mais focadas, e tons de limiarização mais claros (mais próximos de 255) apresentam detecção de gotas desfocadas pois gotas desfocadas possuem tons mais claros em relação às gotas focadas.

Figura 24. Imagens de gotas utilizadas como exemplo para mostrar tons de cinza. (a) Gota Focada (b) Gota Desfocada



(a)



(b)

Os tons de cinza mais próximos de zero (preto) são mais escuros que os tons de cinza mais próximos de duzentos e cinquenta e cinco (branco). A tabela abaixo contém os valores dos pixels utilizados para exemplo.

**Tabela 1. Média dos tons de cinza encontrados nas gotas.**

Sequencia	1	2	3	4	5	6	7	8	9	10
Focada	134	32	0	115	132	125	137	56	38	140
Desfocada	139	135	123	96	103	100	102	107	124	137

Representando graficamente a tabela acima para observar-se a variação dos tons de cinza na gota.

Figura 25. Média dos tons de cinza de gota focada e gota desfocada.

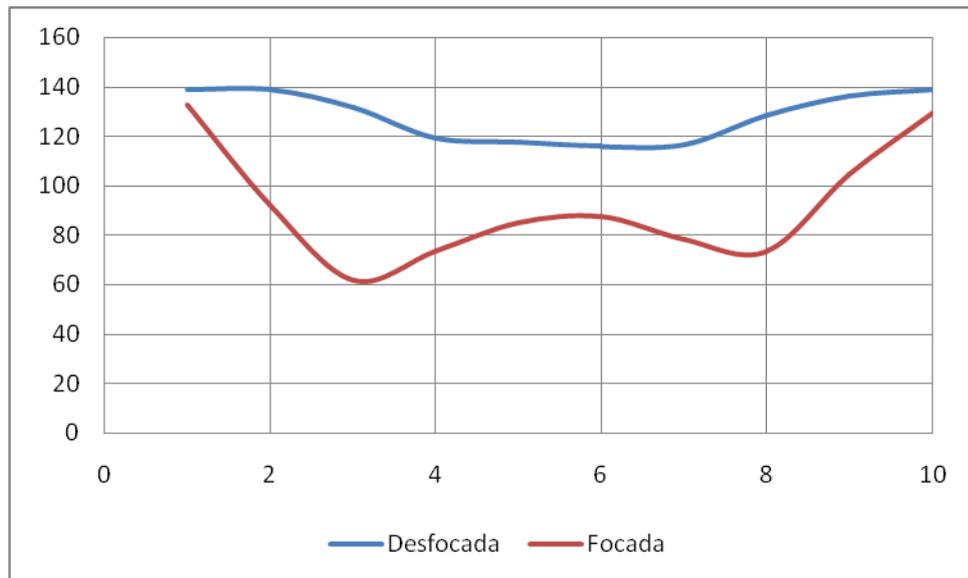


Figura 26. Gráfico 3D dos tons de cinza de uma gota focada.

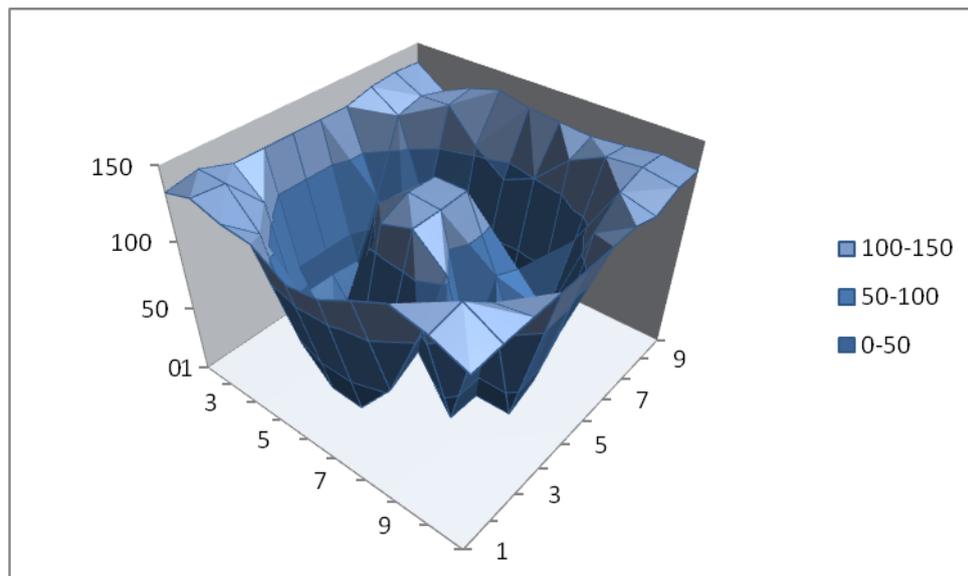
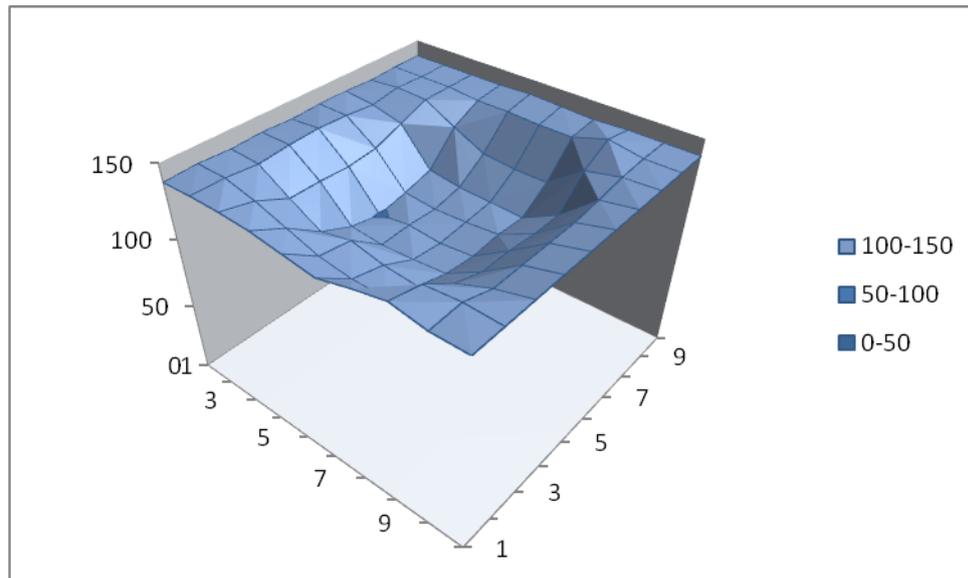


Figura 27. Gráfico 3D dos tons de cinza de uma gota desfocada.



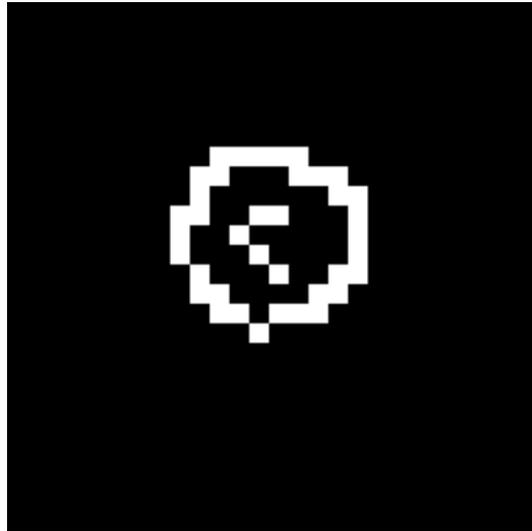
Por meio dos gráficos gerados, conclui-se que gotas focadas possuem tons mais escuros que as gotas desfocadas, e também pode-se observar o reflexo da luz de fundo em seu interior.

O detector de bordas está configurado para limiarizar tons que capturem gotas focadas e descarte gotas desfocadas.

### 3.9 SEGMENTAÇÃO (MORFOLOGIA MATEMÁTICA)

O reflexo da luz de fundo aparente no interior da gota faz com que seja detectado bordas interiores e exteriores de cada gota focada. Com este problema de detectar bordas no interior da gota, devido o reflexo de luz de fundo no centro da gota, aplicou-se então um filtro de morfologia matemática (fechamento) para unir as bordas das gotas.

Figura 28. Imagem das bordas no interior de uma gota.

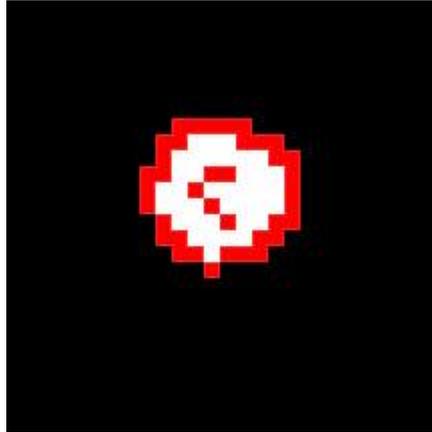


O filtro de fechamento deve unir as bordas da gota e do reflexo no interior da gota. O filtro de morfologia matemática é aplicado utilizando *kernel* 11x11 em formato *rhombus*.

$$\left( \begin{array}{cccccccccccc} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right)$$

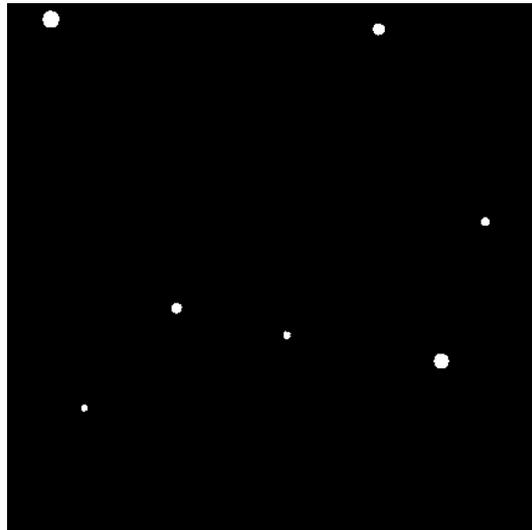
A imagem resultante possui bordas interiores e exteriores conectadas, formando então *pixels* de uma única gota.

Figura 29. Gota com bordas interiores e exteriores conectadas.



Em branco temos os pixels alterados após operação de fechamento, e em vermelho temos os pixels das bordas anteriores do fechamento. O resultado é uma imagem com pixels brancos representando a região de interesse (gotas) e pixels pretos representando *background* (plano de fundo).

Figura 30. Imagem binária das gotas após filtro de detecção e fechamento.



### 3.10 SEGMENTAÇÃO POR AGRUPAMENTO (APRENDIZADO DE MÁQUINA)

Nesta etapa foi determinada a quantidade de gotas na imagem, uma vez a imagem se apresentava de forma das gotas e não era possível determinar a quantidade de gotas e principalmente o seu tamanho.

Uma tentativa de quantificar a quantidade de gotas foi desenvolvida calculando distância Euclidiana entre todos *pixels* brancos, os *pixels* que apresentarem uma distância igual a 1 estão conexos, e são inseridos em um vetor, tendo então um vetor de vetores. Cada posição do vetor representa uma gota, e cada posição representa cada *pixel* da gota.

Este método apresentou problemas na divisão de *pixels* de gotas quando duas gotas estão conectadas, fazendo com que duas ou mais gotas fossem identificadas como uma só.

Outra tentativa foi agrupar as coordenadas dos *pixels* brancos utilizando *K-Means*. Este algoritmo divide os dados em grupos baseados em sua distância Euclidiana, porém para execução do algoritmo, é necessário informar a quantidade de grupos em que os dados serão divididos. Cada grupo deve representar uma gota, e cada elemento do *cluster* um pixel da gota, no entanto, nesta etapa não é possível saber a quantidade de gotas na imagem para executar o agrupamento.

Testes foram realizados re-executando *K-Means* incrementando automaticamente a quantidade de grupos de acordo com a distância entre pontos dentro de um grupo. Quando duas gotas são inseridas em um só grupo, a distância entre os pontos é alta, e então deve-se aumentar a quantidade de grupos para dividir este grupo em dois.

Este método tornou-se inviável quando o tempo de execução do agrupamento aumentou conforme a quantidade de vezes que o *KMeans* deve ser re-executado para encontrar o número ideal de grupos, o tempo de execução variou entre 5 e 25 segundos para encontrar um bom número de grupos.

Pensando em agrupar por aglomerado de *pixels*, a solução foi a utilização de um algoritmo de agrupamento espacial baseado em densidade de aplicações com ruído em inglês *Density Based Spatial Clustering of Application with Noise* (DBSCAN), este algoritmo realiza o agrupamento dos dados sem ser necessário informar a quantidade de grupos para execução, é necessário apenas um raio e uma quantidade mínima de elementos que devem estar dentro deste raio.

O DBSCAN foi executado agrupando as coordenadas dos *pixels* brancos, utilizando um raio de 3 e onde ocorram pelo menos 2 pontos (*pixels*) dentro deste raio. O DBSCAN demonstrou bons resultados em uma perfeita seleção dos pixels das gotas e um tempo de execução menor que o do *KMeans*, demorando de 10 a 50 milissegundos de execução.

### **3.11 EXTRAÇÃO DE CARACTERÍSTICAS**

A execução do DBSCAN resulta em grupos de *pixels*, onde a quantidade de grupos representa a quantidade de gotas, e a quantidade de elementos em cada grupo representa a quantidade de *pixels* dentro de cada gota, sendo possível então calcular a quantidade de gotas em uma imagem e seus respectivos tamanhos.

### 3.12 INTERPRETAÇÃO

Nesta etapa foi calculado o tamanho das gotas encontradas, como as gotas não apresentam um formato regular, utiliza-se um método que aproxima a gota de uma elipsóide, e então calcula-se seu tamanho.

Este método consiste em medir dois diâmetros da gota, um horizontal (a) e um vertical (b), onde a raiz quadrada da multiplicação dos dois diâmetros resulta no diâmetro aproximado da gota como uma circunferência perfeita, então multiplica-se por 20, que é o fator de escala de acordo com uma medida capturada por meio de uma imagem de uma régua, onde 1 milímetro foi representado por 50 *pixels*, é necessário que o resultado esteja em micrometros, um milímetro possui mil micrometros, dividindo então 1.000 por 50 sabemos que um *pixel* representa 20 micrômetros.

Equação 1. Cálculo do diâmetro da gota.

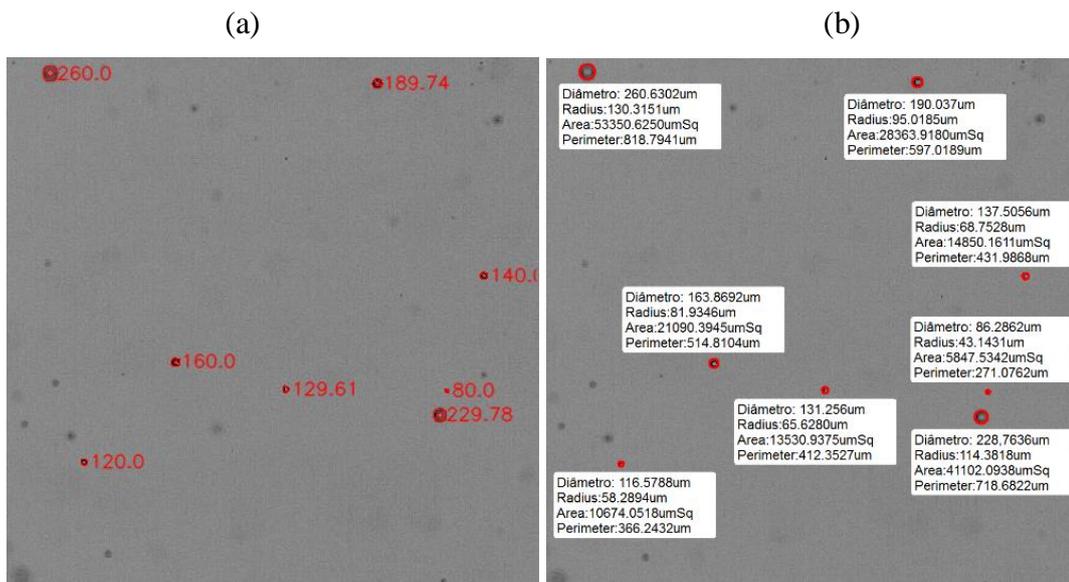
$$\sqrt{a * b} * 20$$

Utilizando esta equação é possível obter com precisão o valor do diâmetro da gota em micrometros.

## 4. VALIDAÇÃO

A validação do método proposto foi realizada comparando resultados obtidos com resultados do *software* de mensuração de tamanhos de objetos ISCapture 2.2.1 da empresa Scienon Technology Co. Ltd., conforme exemplificado nas Figura 31a (método proposto), 31b (ISCapture).

Figura 31 – Resultados obtidos da validação do método.



Os resultados dos diâmetros das gotas observados nas Figura 31a e Figura 31b apresentaram um coeficiente de correlação de Pearson ( $p \leq 0,05$ ) de 0,99 mostrando alta correlação. Cabe destacar que este aparelho foi utilizado por Oliveira et al. (2019) validando resultados de quantificação do diâmetro de gotas.

## 5. CONSIDERAÇÕES FINAIS

Este trabalho apresentou um método capaz de determinar precisamente o tamanho de gotas em imagens, utilizando técnicas de processamento de imagem como: filtro de suavização Gaussiana, filtro de detecção de bordas *Canny Edge*, operação “fechamento” de Morfologia Matemática e algoritmo de agrupamento DBSCAN.

O filtro de suavização Gaussiana obteve destaque por suavizar os ruídos e gotas sem marcar as bordas com a matriz de convolução utilizada, a redução dos ruídos e a suavização das bordas auxiliaram em uma boa detecção de bordas de gotas.

O detector de bordas *Canny Edge* satisfaz os três critérios de um bom detector propostos por Canny (1986), as bordas das gotas identificadas permitiram que os ruídos da imagem e as gotas desfocadas fossem desconsiderados e que fosse detectado apenas bordas de gotas focadas.

A operação “fechamento” de Morfologia Matemática realizou a união de pixels de bordas internas e externas das gotas possibilitando que os *pixels* internos das gotas não fossem identificados como gotas.

O algoritmo de agrupamento DBSCAN agrupou os pixels de bordas encontrados, através destes grupos foi possível determinar a quantidade de gotas na imagem e a quantidade de pixels por gotas.

## 6. REFERÊNCIAS

BACHALO, W., & HOUSER, M. **Development of the phase/Doppler spray analyzer for liquid drop size and velocity characterizations.** 20th Joint Propulsion Conference. doi:10.2514/6.1984-1199, 1984.

CANNY, J. **A Computational Approach to Edge Detection**, IEEE Transactions on Pattern Analysis and Machine Intelligence, V. 8, n. 6, pp. 679-698, 1986.

CARVALHO, F., ANTUNIASSI, U., CHECHETTO, R., MOTA A., JESUS, M., CARVALHO, L. **Viscosity, surface tension and droplet size of sprays of different formulations of insecticides and fungicides.** Crop Protection v. 101 p. 19-23, 2017.

CHEN, H., TANG, H., LIU, Y., WANG, H., LIU, G. **Measurement of particle size based on digital imaging technique.** Journal of Hydrodynamics. Hohai University, Nanjing 210098, China, v. 25, n. 2, p. 242-248, 2013.

ESTER, M., KRIEGEL, H., SANDER, J., XU, X. **A density-based algorithm for discovering clusters in large spatial databases with noise**, International Conference on Knowledge Discovery and Data Mining, p. 226-231, Institute of Computer Science, University of Munich, 1996.

FRANCA, J.; CUNHA, J.; ANTUNIASSI, U. **Spectrum and velocity of droplets of spray nozzles with and without air induction.** Eng. Agríc., Jaboticabal, v. 38, n. 2, p. 232-237, 2018.

JACTO, S. A. **Manual técnico sobre orientação de pulverização.** Pompéia, p. 23, Edição 10/99. Código - 957928, 1999.

JESUS, E., COSTA, R. JR., **A Utilização de Filtros Gaussianos na Análise de Imagens Digitais**, Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, V. 3, N. 1, p. 7, 2014.

LEARNED-MILLER, E. G. **Introduction to computer vision**. University of Massachusetts, Amherst, 2011.

LEFEBVRE, A.H. **Atomization and sprays. International Series: Combustion**. New York: Hemisphere Publishing Corporation, 421p, 1989.

MÁRQUEZ, L. **Tecnología para la aplicación de defensivos agrícolas**. In: Congresso Brasileiro de Engenharia Agrícola, 26, Campina Grande. Anais... Palestra, 1997.

MAINI, R. and AGGARWAL H. **Study and comparison of various image edge detection techniques** International Journal of Image Processing (IJIP), vol. 3, pp. 1-11, 2009.

MATTHEWS, G.A. **The application of chemicals for plant disease control**. In: WALLER, J.M.; LENNÉ, J.M.; WALLER, S.J. Plant pathologist's pocketbook. London: CAB, p.345-53, 2002.

MATUO, T. Fundamentos da tecnologia de aplicação de agrotóxicos. In: GUEDES, J.; DORNELESS, S. (Org.) **Tecnologia e segurança na aplicação de agrotóxicos: novas tecnologias**. Santa Maria: Departamento de Defesa Sanitária: Sociedade de Agronomia de Santa Maria, 1998.

MICHALSKI R. S., CARBONELL, J. G., MITCHELL, T. M. **Machine Learning: An Artificial Intelligence Approach**, Springer Science & Business Media, 2013.

SAINI, S., KASLIWAL, B., BHATIA, S. **Comparative study of image edge detection algorithms**, Computing Science and Engineering, Institute of Technology, India, 2010.

NUYTTENS, D., DE SCHAMPHELEIRE, M., VERBOVEN, P., BRUSSELMAN, E., DEKEYSER, D. **Droplet size and velocity characteristics of agricultural sprays**. Transactions of the ASABE 52, 2009.

OLIVEIRA, R. B. **Caracterização Funcional de Adjuvantes em Soluções Aquosas**. Tese (Doutorado em Energia na Agricultura). 2011. 121 f. Universidade Estadual Paulista "Júlio de Mesquita Filho" Faculdade de Ciências Agrônomicas – Botucatu – SP. 2011.

OLIVEIRA, R. B., PRECIPITO, L. M. B., GANDOLFO, M. A., OLIVEIRA, J. V., LUCIO, F. R. **Effect of droplet size and leaf surface on retention of 2,4-D.** Crop Protection, v. 119, p. 97-101, 2019.

RADHIKA, C. **Comparison of Edge Detection Techniques,** [https://www.researchgate.net/publication/297736749\\_Comparison\\_of\\_Edge\\_Detection\\_Techniques](https://www.researchgate.net/publication/297736749_Comparison_of_Edge_Detection_Techniques), Setembro, 2013.

RAUTARAY, S. S., & AGRAWAL, A. **Vision based hand gesture recognition for human computer interaction: a survey.** Artificial Intelligence Review, v. 43, n. 1, p. 1-54, 2015.

ROSEBROCK, A. **Zero-parameter, automatic Canny edge detection with Python and OpenCV,** disponível em <https://www.pyimagesearch.com/2015/04/06/zero-parameter-automatic-canny-edge-detection-with-python-and-opencv/>, 2015.

SHAPIRO, L. G., STOCKMAN, G. **Computer Vision,** Prentice Hall, 2000.

TEIXEIRA, M. **Influencia del volumen de caldo y de la uniformidad de distribución transversal sobre la eficacia de la pulverización hidráulica.** 1997. 310 f. Tese (Doutorado em Engenharia Rural) - Escuela Técnica Superior de Ingenieros Agrónomos, Universidad Politécnica de Madrid, Madrid, 1997.

QUEIROZ, J. E. R, GOMES, H. M., **Introdução ao Processamento Digital de Imagens,** Revista RITA: Instruções para preparação de documentos em Word, vol. 8, n. 1, 2001.

WANG, Z., KIEU, H., NGUYEN, H., & LE, M. **Digital image correlation in experimental mechanics and image registration in computer vision: similarities, differences and complements.** Optics and Lasers in Engineering, v. 65, n. 5, p. 18-27, 2015.