

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO**

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA**  
**ANÁLISE DE IMAGENS AÉREAS PARA A**  
**AGRICULTURA DE PRECISÃO**

**ADRIANO GRAEBIN**

**BLUMENAU**  
**2017**

**ADRIANO GRAEBIN**

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA  
ANÁLISE DE IMAGENS AÉREAS PARA A  
AGRICULTURA DE PRECISÃO**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Ciência da Computação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Prof. Aurélio Faustino Hoppe, Mestre – Orientador

**BLUMENAU  
2017**

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA  
ANÁLISE DE IMAGENS AÉREAS PARA  
A AGRICULTURA DE PRECISÃO**

Por

**ADRIANO GRAEBIN**

Trabalho de Conclusão de Curso aprovado  
para obtenção dos créditos na disciplina de  
Trabalho de Conclusão de Curso II pela banca  
examinadora formada por:

Presidente: \_\_\_\_\_  
Prof. Aurélio Faustino Hoppe, Mestre – Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Matheus Luan Krueger, Mestre – FURB

Membro: \_\_\_\_\_  
Prof. Mauro Marcelo Mattos, Doutor – FURB

Blumenau, 04 de julho de 2017

Dedico esse trabalho aos que lutam pela única estrutura para o futuro da nossa sociedade, o desenvolvimento sustentável.

## **AGRADECIMENTOS**

Agradeço aos meus pais, Clari e Dirce, pelo amor, incentivo e apoio incondicional.

Ao meu orientador, Aurélio Faustino Hoppe, pelo apoio no desenvolvimento deste trabalho.

Ao corpo docente do departamento de Computação da FURB, do Instituto Federal Catarinense e da Technische Universität Dortmund pelos ensinamentos durante minha jornada acadêmica.

A todos que direta ou indiretamente tornaram este trabalho possível.

*We can only see a short distance ahead, but  
we can see plenty there that needs to be done.*

Alan Turing

## RESUMO

A tecnologia da informação já está presente no campo e tem papel fundamental na agricultura moderna e globalizada, denominada “Agricultura 4.0”. A crescente demanda por uma produção agrícola eficiente e sustentável requer o desenvolvimento de novas soluções que possibilitem o setor “produzir mais com menos”. Nesse sentido, este trabalho apresenta o desenvolvimento de um protótipo de software que possibilite a análise de imagens aéreas de alta resolução no sentido de avaliar a variabilidade espacial de uma plantação. Para tal, imagens capturadas por satélites ou Veículos Aéreos Não Tripulados (VANTs) são processadas a partir de técnicas de reconhecimento de padrões, com base em amostras – ou fragmentos da imagem – informadas pelo usuário. O desenvolvimento contemplou uma aplicação web com a arquitetura cliente-servidor, mais especificamente o *framework* JavaServer Faces (JSF) e uma base de dados NoSQL, enquanto o processamento das imagens envolveu a ferramenta MATLAB. Utilizando conceitos alinhados à Agricultura de Precisão, os mapas gerados e áreas calculadas pelo protótipo podem ser utilizadas por gestores da produção agrícola para a análise da variabilidade espacial de uma plantação. A assertividade dos resultados apresentou dependência direta com a qualidade das imagens de entrada e das amostras selecionadas pelo usuário.

Palavras-chave: Reconhecimento de padrões. Agricultura de precisão. Agricultura 4.0.

## **ABSTRACT**

Information technology already takes place in farming and plays a key role in modern and globalized agriculture, called "Farming 4.0". The growing demand for an efficient and sustainable agricultural production requires the development of new solutions that enable the sector to "produce more with less". In this sense, this work presents the development of a software prototype that allows the analysis of high resolution aerial images in order to evaluate the spatial variability of a field. To this end, images captured by satellites or Unmanned Aerial Vehicles (UAVs) are processed with pattern recognition techniques based on samples – or fragments of the image – defined by the user. The development included a web application with a client-server architecture, more specifically the JavaServer Faces (JSF) framework and a NoSQL database, while the image processing method involved MATLAB. Using concepts aligned to Precision Agriculture, the maps and areas calculated by the prototype can be used by agricultural production managers to analyze the spatial variability of a plantation. The obtained results have direct dependence on the quality of the input images and the samples defined by the user.

**Key-words:** Pattern recognition. Precision agriculture. Farming 4.0.



## LISTA DE FIGURAS

Figura 1 – Imagem aérea de um campo produtivo antes e depois de processada .....	13
Figura 2 – Etapas da Agricultura de Precisão .....	15
Figura 3 – Causas da variabilidade e grau de dificuldade para sua intervenção .....	16
Figura 4 – Classificação Não-Supervisionada.....	19
Figura 5 – Esquema de operação.....	21
Figura 6 – Resultados alcançados.....	22
Figura 7 – Classificações identificadas .....	23
Figura 8 – Diagrama de casos de uso .....	25
Figura 9 – Diagrama de atividades .....	27
Figura 10 – Página principal.....	28
Figura 11 – Seleção de uma amostra .....	29
Figura 12 – Pré-processamento .....	30
Figura 13 – Limites superior e inferior de intensidade.....	31
Figura 14 – Comparativo das faixas de intensidade .....	32
Figura 15 – Intensidade de uma amostra .....	33
Figura 16 – Comparativo entre limites e amostra.....	34
Figura 17 – Resultados na página de análise de imagens.....	35
Figura 18 – Imagem virtual e resultados obtidos .....	36
Figura 19 – Mapa com região divergente.....	43

## LISTA DE QUADROS

Quadro 1 – Matriz de rastreabilidade de RF e UC .....	24
Quadro 2 – Matriz de rastreabilidade de RNF e UC .....	24
Quadro 3 – Pré-processamento.....	30
Quadro 4 – Cálculo dos limites inferior e superior .....	31
Quadro 5 – Definição da intensidade das amostras .....	32
Quadro 6 – Definição dos limites de intensidade .....	33
Quadro 7 – Resultados alcançados (Continua).....	37
Quadro 8 – Comparativo entre os trabalhos correlatos e o trabalho atual.....	41
Quadro 9 – Dependência Maven .....	46
Quadro 10 – Notação das funções .m.....	46
Quadro 11 – Biblioteca matlabcontrol.....	47

## **LISTA DE ABREVIATURAS E SIGLAS**

AP – Agricultura de precisão

JSF – JavaServer Faces

NDVI – Índice de Vegetação por Diferença Normalizada

noSQL – Non-Structured Query Language

RF – Requisitos funcionais

RGB – Red, green and blue

RNF – Requisitos não-funcionais

TI – Tecnologia da Informação

UAVs – Unmanned Aerial Vehicles

UC – Casos de uso

VANTs – Veículos Aéreos Não Tripulados

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>12</b>
1.1 OBJETIVOS.....	13
1.2 ESTRUTURA.....	13
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>15</b>
2.1 AGRICULTURA DE PRECISÃO.....	15
2.2 RECONHECIMENTO DE PADRÕES .....	17
2.3 TRABALHOS CORRELATOS.....	20
2.3.1 Crops diagnosis using digital image processing and precision agriculture technologies	20
2.3.2 Uma arquitetura para sistemas espaço-temporais baseados na web para Agricultura de Precisão .....	21
2.3.3 Weed mapping in early-season maize fields using object-based analysis of unmanned aerial vehicle (UAV) images.....	23
<b>3 DESENVOLVIMENTO DO PROTÓTIPO.....</b>	<b>24</b>
3.1 REQUISITOS.....	24
3.2 ESPECIFICAÇÃO .....	24
3.2.1 Técnicas e ferramentas utilizadas.....	24
3.2.2 Diagrama de casos de uso .....	25
3.2.3 Fluxo principal do protótipo .....	26
3.3 ANÁLISE DOS RESULTADOS .....	35
3.3.1 Testes com imagens virtuais .....	36
3.3.2 Testes com imagens áreas .....	36
3.3.3 Comparativo dos resultados com os trabalhos correlatos .....	40
<b>4 CONCLUSÕES.....</b>	<b>42</b>
4.1 LIMITAÇÕES.....	42
4.2 EXTENSÕES .....	43
<b>REFERÊNCIAS .....</b>	<b>44</b>
<b>APÊNDICE A – CONFIGURAÇÃO DA API MATLABCONTROL .....</b>	<b>46</b>

## 1 INTRODUÇÃO

A produção agrícola está redirecionando seu foco de quantidade para qualidade e sustentabilidade. Segundo López et al. (2015), nos métodos convencionais de produção agrícola os tratamentos para problemas relacionados à baixa produtividade são feitos de forma homogênea. Isto é, não existe segmentação por áreas afetadas, por exemplo, por pragas ou degradação do solo – a dosagem de herbicidas ou fertilizantes é aplicada uniformemente por toda a lavoura. Além disso, na agricultura convencional a identificação de anomalias dentro de uma plantação (como pragas em um setor específico) é realizada manualmente, sem qualquer tipo de automatização ou processo que facilite a análise (BERNARDI et al., 2014).

Dessa forma, o interesse do setor agrícola por ferramentas que auxiliem a tomada de decisão na gestão de produção exige o aperfeiçoamento dos sistemas de informação atuais e a criação de novas soluções (JENSEN, 2000). Já Antonopoulou et al. (2010) afirmam que com base em dados coletados, armazenados e processados por um sistema de apoio à decisão, agrônomos, legisladores e agricultores podem tomar potenciais ações na gestão da produção agrícola.

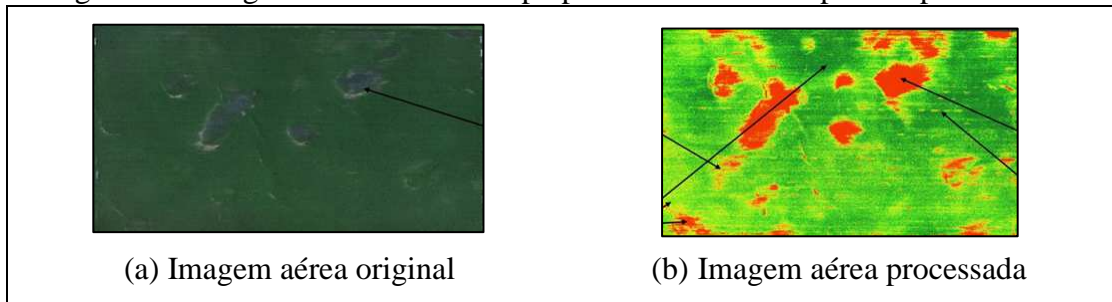
Murakami (2006) aponta que existem alguns sistemas voltados para a Agricultura de Precisão (AP). No âmbito acadêmico, o autor concluiu que os sistemas, em sua maioria, são pouco abrangentes e que são projetados para problemas específicos da AP. No âmbito comercial, os sistemas são mais abrangentes e permitem o tratamento de dados gerados na etapa de coleta de dados. Entretanto, conclui que estes sistemas não atendem plenamente às necessidades dos agricultores em termos de funcionalidades, custos e capacidade de integração com outros sistemas.

Sugiura et al. (2003) indicam que é possível extrair informações relevantes da produção agrícola a partir de imagens de satélite. Para Peña et al. (2013), a resolução deste tipo de imagem dificulta a determinação de características do solo e da plantação em alguns cenários, contudo esta problemática tem sido resolvida com a introdução da nova geração de Veículos Aéreos Não Tripulados (VANTs). Com eles é possível capturar imagens de altíssima resolução, já que podem ser operados em altitudes muito baixas – algo impraticável nos outros métodos.

O uso do processamento deste tipo de imagem mostra-se muito oportuno aos objetivos da AP, pois é caracterizado por resultados precisos que apresentam uma assertividade entre 86% e 95%, dependendo dos algoritmos utilizados e da qualidade das imagens (VIBHUTE;

BODHE, 2012). A Figura 1 apresenta um exemplo de processamento de imagem aérea no âmbito da Agricultura de Precisão.

Figura 1 – Imagem aérea de um campo produtivo antes e depois de processada



Fonte: adaptado de VanderLeest et al. (2016).

Observa-se que a representação visual do item (b) da Figura 1 é mais relevante. Nela, é possível perceber áreas demarcadas em vermelho que apresentam problemas, as quais não são claramente identificadas no item (a) da Figura 1.

Desse modo, a análise de imagens por meio de um software para a AP, com mecanismos simples e de clara utilização, oferece um grande facilitador para a gestão da produção agrícola. Diante deste cenário, o presente trabalho apresenta o desenvolvimento de um protótipo capaz de realizar o processamento de imagens aéreas de alta resolução no sentido de avaliar a variabilidade espacial de uma plantação.

## 1.1 OBJETIVOS

O objetivo deste trabalho é desenvolver um protótipo de software que realize o processamento de imagens aéreas de alta resolução no sentido de avaliar a variabilidade espacial de uma plantação.

Os objetivos específicos são:

- a) identificar padrões a partir de amostras informadas pelo usuário dentro de uma imagem aérea de uma plantação;
- b) gerar um mapa identificando áreas de recorrência de cada uma das amostras informadas;
- c) mensurar os resultados envolvendo a área de recorrência de cada uma das amostras informadas em percentual.

## 1.2 ESTRUTURA

Este trabalho está dividido em quatro capítulos. O primeiro capítulo apresenta a motivação para este trabalho e os objetivos do mesmo. O segundo capítulo apresenta a fundamentação teórica que embasa o desenvolvimento do trabalho. No terceiro capítulo é

descrito o detalhamento da implementação do método e os resultados dos testes realizados para validar a implementação. Por fim, o quarto capítulo contempla as conclusões e limitações, assim como sugestões para possíveis extensões deste trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

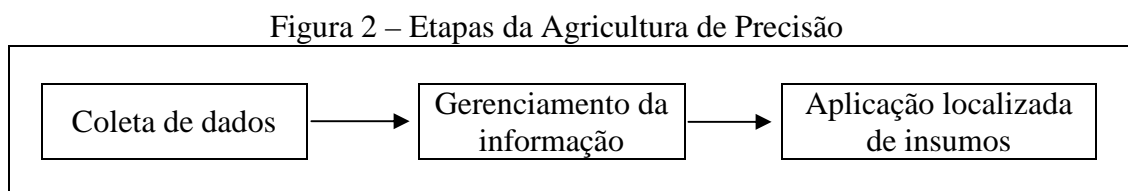
Este capítulo aborda os assuntos e técnicas utilizadas para o desenvolvimento do protótipo. Nas próximas seções serão abordados os assuntos: Agricultura de Precisão e Reconhecimento de Padrões.

### 2.1 AGRICULTURA DE PRECISÃO

Segundo o National Research Council (1997), a Agricultura de Precisão se trata de uma estratégia de gestão que faz uso da Tecnologia da Informação (TI) para que dados coletados de uma safra suportem a tomada de decisão na produção agrícola. A AP visa ainda a maximização da produtividade acompanhada da redução de custos e impactos ambientais na produção.

Além disso, a AP proporciona um gerenciamento da produção muito mais eficaz do que os métodos convencionais de produção, nos quais não se faz uso de qualquer tipo de TI para verificar a variabilidade da produção dentro de uma mesma área produtiva. Uma grave consequência desta desconsideração é a aplicação indevida e generalizada de insumos, como fertilizantes e agrotóxicos. Assim, perdas e impactos ambientais desnecessários são recorrentes (NATIONAL RESEARCH COUNCIL, 1997).

Saraiva (2003) divide o processo da AP em três etapas principais, conforme ilustrado na Figura 2.



Fonte: adaptado de Saraiva (2003).

A primeira etapa consiste na obtenção dos dados que suportam todo o processo. Saraiva (2003) afirma que a coleta de dados tem como objetivo obter informações relevantes para a produção, digitalizando parâmetros como a presença de pragas, ervas daninhas, condição do solo, dentre outros.

O gerenciamento da informação consiste em diferentes operações (SARAIVA, 2003):

- a) tratar e corrigir erros nos dados coletados, ocasionados por métodos e/ou equipamentos imprecisos;
- b) processar os dados coletados para avaliar e quantificar a variabilidade de um parâmetro;

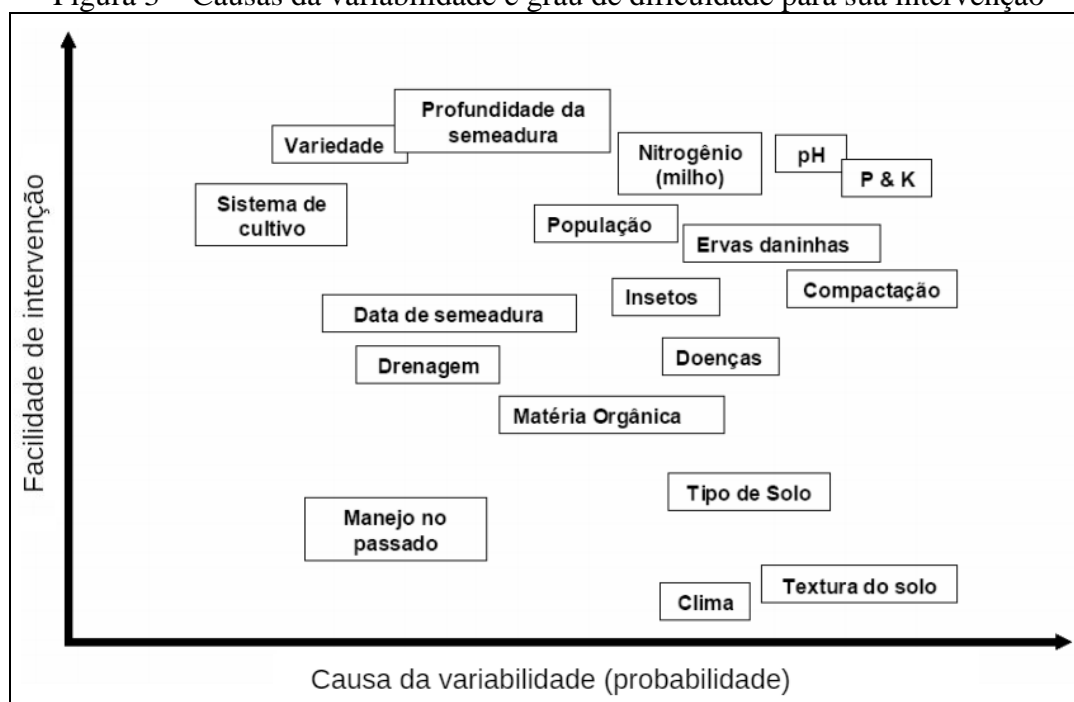


- c) buscar relações de causa/efeito, relacionando a variabilidade da produção (colheita) com a dos fatores influentes (solo, insumos, pragas, etc.);
- d) propor estratégias a partir dos resultados obtidos no item anterior, expressos através de mapas de aplicação de insumos em taxa variável em campo, para atingir as metas de gerenciamentos desejadas – sejam relacionadas à produtividade, redução de insumos aplicados ou mesmo a redução de impactos ambientais.

Por fim, na terceira etapa, é realizada a aplicação de insumos agrícolas com taxas diferenciadas a partir das conclusões alcançadas. Os mapas resultantes da etapa anterior devem mostrar a variação espacial de variáveis da área sendo avaliada, como: fertilidade do solo, incidência de ervas daninhas, incidência de pragas ou doenças, umidade, acidez, estrutura e drenagem do solo, etc. Com isso, o produtor pode manipular e avaliar as informações a fim de criar uma prescrição técnica para o tratamento de uma área (SARAIVA, 2003).

As correlações entre efeito e as possíveis causas da variabilidade espacial em uma lavoura têm sido exaustivamente estudadas por muitos pesquisadores e os valores, via de regra, são baixos, e por vezes, conflitantes (DOMINGUES JUNIOR; INAMASU, 2010). A Figura 3 apresenta um exemplo de um caso de análise de fatores envolvidos na variabilidade espacial de uma lavoura com seus diferentes graus de probabilidade e de facilidade de correção.

Figura 3 – Causas da variabilidade e grau de dificuldade para sua intervenção



Fonte: adaptado de Domingues Junior e Inamasu (2010).

Lamparelli, Rocha e Borghi (2001) observam que a primeira e terceira etapa tratam de problemas delimitados e, que quando utilizados equipamentos e técnicas adequadas, alcançam resultados assertivos, enquanto que a etapa que gerencia a informação é muito mais crítica. Nela, a escolha da melhor forma para gerar informações consistentes – seja com algoritmos, índices, mapas, etc. – demanda análises aprofundadas e vastos conhecimentos especializados.

Nesse sentido, Bernardi et al. (2014) afirmam que a gestão da variabilidade traz subsídios para melhorar o entendimento da lavoura, mas nem sempre retorno econômico. A melhoria nos processos relacionados à AP é contínua, de forma que se um produtor detectar variabilidade espacial causada por máquinas e ele aprimorar o seu uso, o próximo ciclo pode ser o de melhorar o uso de fertilizantes e assim por diante. Segundo o autor, analisar a causa e identificar a variabilidade é fundamental nesse processo.

Dessa forma, os métodos de AP mostram-se custosos, mas possibilitam, além de bons resultados, um processo agrícola mais racional, responsável e rastreável, demandante de mão de obra qualificada e, conseqüentemente, de desenvolvimento sustentável. A aplicação destes métodos tende a se tornar cada vez mais comum nas propriedades rurais, uma vez que se tornem viáveis e acessíveis aos interesses dos produtores agrícolas.

## 2.2 RECONHECIMENTO DE PADRÕES

O Reconhecimento de Padrões é o campo da ciência que tem por objetivo a classificação de objetos em um determinado número de categorias ou classes a partir da observação de suas características (THEODORIDIS; KOUTROUMBAS, 2007).

Gomes (2007) aponta que definir ou categorizar um objeto em uma imagem pode ser uma tarefa complexa. Considerando que um objeto é qualquer porção da imagem que tenha pelo menos uma característica mensurável que a diferencie do restante da imagem, regiões podem ser categorizadas como objetos por terem determinado tamanho ou até mesmo por apresentarem uma certa forma.

Os pixels de uma imagem também podem ser considerados objetos. Gonzalez e Woods (2007) afirmam que uma imagem pode ser definida como uma função bidimensional,  $f(x, y)$ , onde  $x$  e  $y$  são coordenadas espaciais. A amplitude de  $f$  em qualquer par de coordenadas  $(x, y)$  é chamada de “intensidade” da imagem naquele ponto, correspondendo a um pixel. Levando em consideração que os pixels podem apresentar diferentes cores ou intensidades em uma imagem, apesar da forma e tamanho uniformes, estes podem ser classificados como quaisquer outros tipos de objetos.

A classificação de pixels em imagens constitui-se em um procedimento de segmentação. É largamente utilizada em imagens de Sensoriamento Remoto, o que engloba também imagens para a AP (GOMES, 2007).

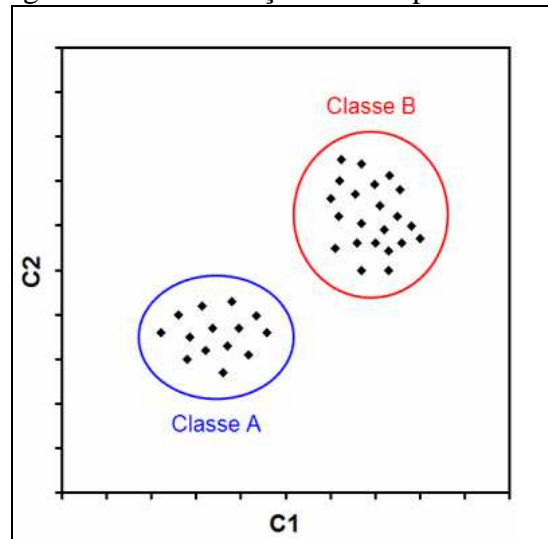
Barbu (2003) afirma que a tarefa de segmentação de imagens consiste na divisão de uma imagem de entrada em diferentes objetos denominados “segmentos da imagem”. Em cada um dos segmentos, todos os pixels apresentam uma característica em comum, chamada de “critério de similaridade”. Neste estudo, o critério de similaridade levou em consideração os valores do sistema de cores Red, Green and Blue (RGB) de cada um dos pixels da imagem em análise.

Segundo Barbu (2003) há uma ampla variedade de métodos de segmentação de imagens, não havendo uma solução geral para a tarefa de segmentação. Cada tipo de imagem requer uma metodologia de segmentação distinta, visto que em cada caso diferentes aspectos devem ser levados em consideração, como cor e/ou textura.

Conforme Gomes (2007), o processo de classificação funciona da seguinte maneira: cada pixel constitui-se de fato em um objeto previamente classificado, de modo que uma imagem forme o conjunto de treinamento. Em seguida, escolhe-se um espaço de características e um classificador adequados. Finalmente, medidas as características dos padrões de cada classe, o sistema poderá classificar pixels de imagens desconhecidas, ou seja, segmentá-las.

Barbu (2003) apresenta uma abordagem de segmentação não-supervisionada baseada no procedimento de segmentação denominado “crescimento de regiões”. A Figura 4 mostra um exemplo de classificação não-supervisionada, onde os padrões dos objetos são desconhecidos em um espaço de características bidimensional. Segundo Gomes (2007), um procedimento de classificação não-supervisionada procura por objetos similares e os agrupa em classes. No exemplo em questão, os objetos foram divididos entre duas classes distintas (classe A e classe B), já que apresentaram características semelhantes.

Figura 4 – Classificação Não-Supervisionada



Fonte: adaptado de Gomes (2007).

Barbu (2003) define em seu método  $K$  diferentes classes de segmentação, denominadas  $C1, C2, \dots, CN$  que têm como característica o padrão  $V$  definido pelo usuário. O Quadro 1 apresenta o pseudocódigo do algoritmo.

Quadro 1 – Abordagem de segmentação não-supervisionada proposta por Barbu (2003)

```

1. For i = 1 to K do
2.    $C_i = \{V(x_i)\};$ 
3.    $C = \{\};$  index = 1; {C - helping set},
4. End
5. m = number of  $C_i$ 's (initially N);
6. For i = 1 to m - 1
7.   For j = 1 to m
8.     If  $d(\text{mean}(C_i), \text{mean}(C_j)) = \min(\text{mean}(C_{ij}), \text{mean}(C_{j1}))$  {
                                     smallest overall dist.} then
9.        $C(\text{index}) = C_i \cup C_j$ ; {region merging}
10.      inc(index);
11.    End
12.  End
13. End

```

Fonte: adaptado de Barbu (2003).

No método de Barbu (2003) as classes (ou critérios para definição cada segmento) são inseridas interativamente pelo usuário. Além disso, a cada *loop* ocorre atribuição de uma nova região a um segmento, levando em consideração a função classificadora definida.

Gomes (2007) aponta que, de forma geral, pode-se dividir os classificadores em dois tipos principais:

- a) *conexionistas* (redes neurais): constituem classificadores complexos, inspirados na estrutura neural de organismos inteligentes, que aprendem por experiência;
- b) *estatísticos*: os objetos do conjunto de treinamento são utilizados para estimar certos parâmetros estatísticos de cada classe.

Os classificadores estatísticos são mais adequados quando os objetos nas várias classes estão distribuídos no espaço de características de acordo com topologias simples e preferencialmente conhecidas. Por outro lado, as redes neurais possuem capacidade de se ajustar a qualquer topologia de classes. Assim, as redes neurais são mais indicadas para os casos onde há classes com distribuições malcomportadas (GONZALEZ; WOODS, 2007).

As técnicas de pré-processamento, esquema de representação e método para a tomada de decisão adequada, dependem do domínio do problema. Um problema bem definido e suficientemente detalhado, onde se tem pequenas variações intra-classes e grandes variações inter-classes, produzirá representações compactas de padrões e consequentemente a estratégia de tomada de decisão será simplificada. Aprender, a partir de um conjunto de exemplos (conjunto de treinamento), é um atributo importante desejado na maioria dos sistemas (BIANCHI, 2006).

## 2.3 TRABALHOS CORRELATOS

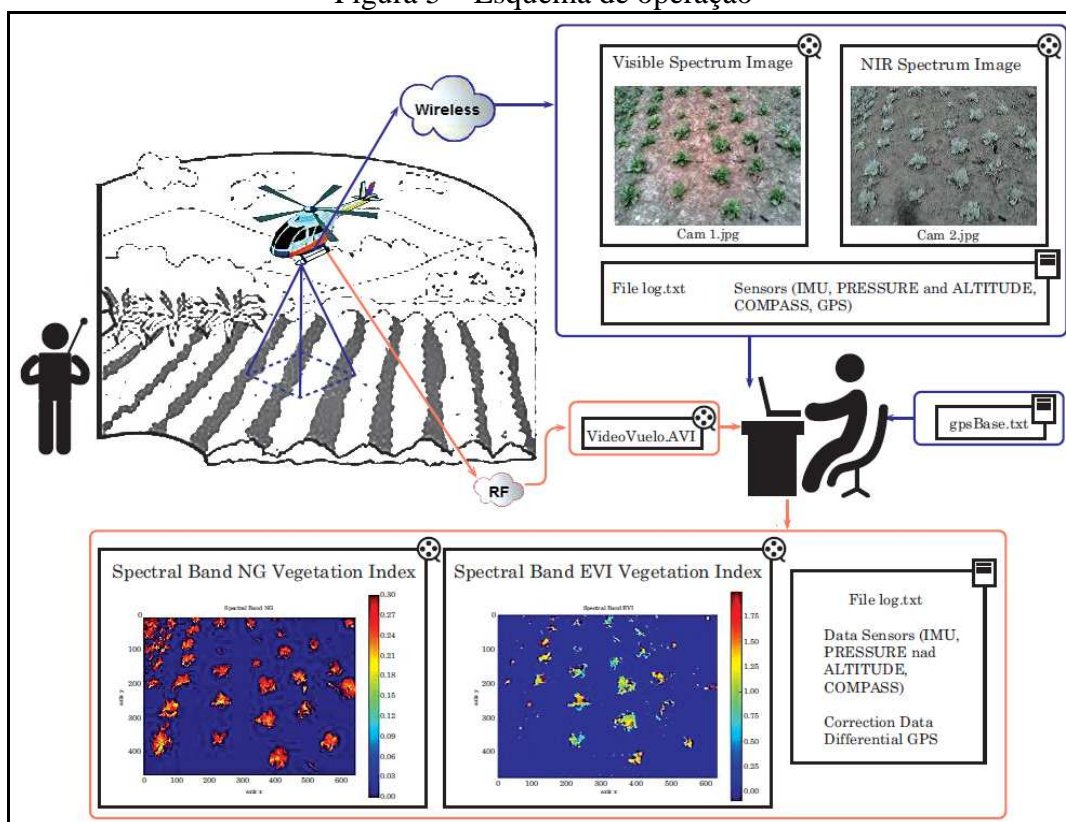
Nesta seção são apresentados trabalhos que apresentam semelhança com os principais objetivos do trabalho proposto. López et al. (2015) desenvolveram um sistema para captura e processamento de imagens para a AP. Já Ribeiro Júnior (2007) propôs uma arquitetura de software para sistemas web voltados para a AP. Por fim, Peña et al. (2013) implementaram um sistema para captura, processamento e geração de mapas de pragas na produção agrícola.

### 2.3.1 Crops diagnosis using digital image processing and precision agriculture technologies

López et al. (2015) apresentam o desenvolvimento de um sistema capaz de adquirir e analisar dados para o diagnóstico da produção agrícola a partir da captura de imagens aéreas. López et al. (2015), engenheiros da Universidade Pedagógica e Tecnológica da Colômbia (UPTC), visaram em seu trabalho a mensuração da variabilidade da produção em um terreno a fim de minimizar custos operacionais na produção.

Na Figura 5 é possível verificar o esquema de funcionamento do sistema, onde se pode ver o passo-a-passo do processo de diagnóstico da produção. Após a captura e carregamento dos dados no software desenvolvido, as imagens capturadas são processadas a partir de um algoritmo que permite a caracterização e diagnóstico de uma unidade produtiva.

Figura 5 – Esquema de operação



Fonte: López et al. (2015).

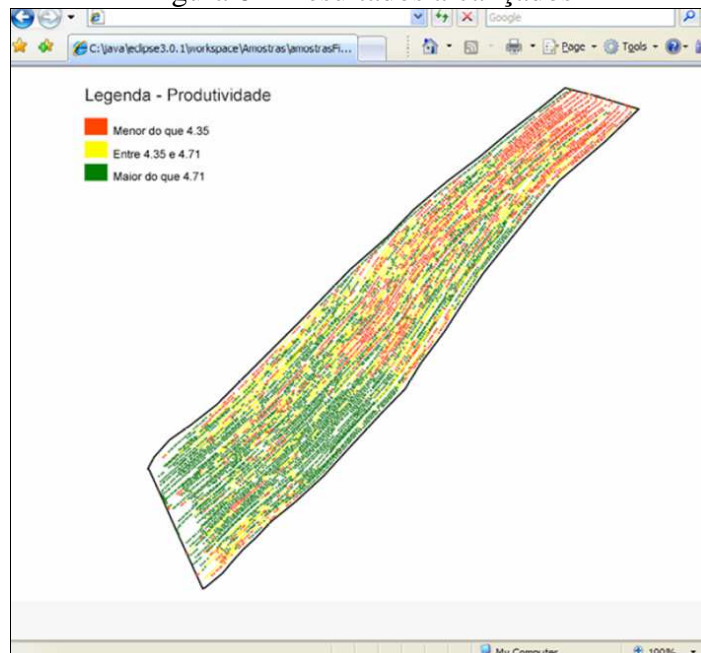
Para o desenvolvimento do projeto foram utilizadas diferentes linguagens de marcação e programação (HyperText Markup Language (HTML), JavaScript, Node.js, C++ e Python). Os resultados alcançados mostram que alguns dos índices utilizados destacaram-se quanto à relevância das informações apresentadas. Contudo, todos os índices permitiram a quantificação de informações importantes em diferentes cenários. Os autores sugerem como extensão a inclusão de índices para outras situações, como doenças, pestes, irrigação e nutrição para culturas específicas.

### 2.3.2 Uma arquitetura para sistemas espaço-temporais baseados na web para Agricultura de Precisão

Ribeiro Júnior (2007) apresenta, como estudo de caso, a implementação de um componente para apresentação de mapas de produtividade com interface web. O autor propõe uma arquitetura de software baseada na plataforma Java Enterprise Edition (JEE) que procura solucionar algumas questões envolvidas no processo de desenvolvimento de software, essencialmente para aplicações no âmbito da agricultura de precisão. Os cenários de arquitetura definidos permitem atender diversas necessidades em aplicações espaço temporais para Web.

O autor também apresenta recomendações de uso e uma análise qualitativa dos cenários arquiteturais, considerando atributos como escalabilidade, portabilidade, simplicidade e extensibilidade. A partir da Figura 6 é possível verificar os resultados extraídos a partir de um software de AP desenvolvido neste *framework*. A produtividade de um campo é identificada por região, de acordo com a legenda.

Figura 6 – Resultados alcançados



Fonte: Ribeiro Júnior (2007).

Embora o projeto de Ribeiro Júnior (2007) não trate diretamente do processamento digital de imagens, este mostra-se eficiente para o desenvolvimento de aplicações no escopo da Agricultura de Precisão, pois oferece ferramentas que dispensam um alto nível de conhecimentos especializados na AP. Desta forma, utilizando-se dos padrões arquiteturais apresentados é claro o ganho na qualidade do produto, já que muitos dos conhecimentos adquiridos já foram testados em experimentos prévios e implementados neste *framework* (RIBEIRO JÚNIOR, 2007).

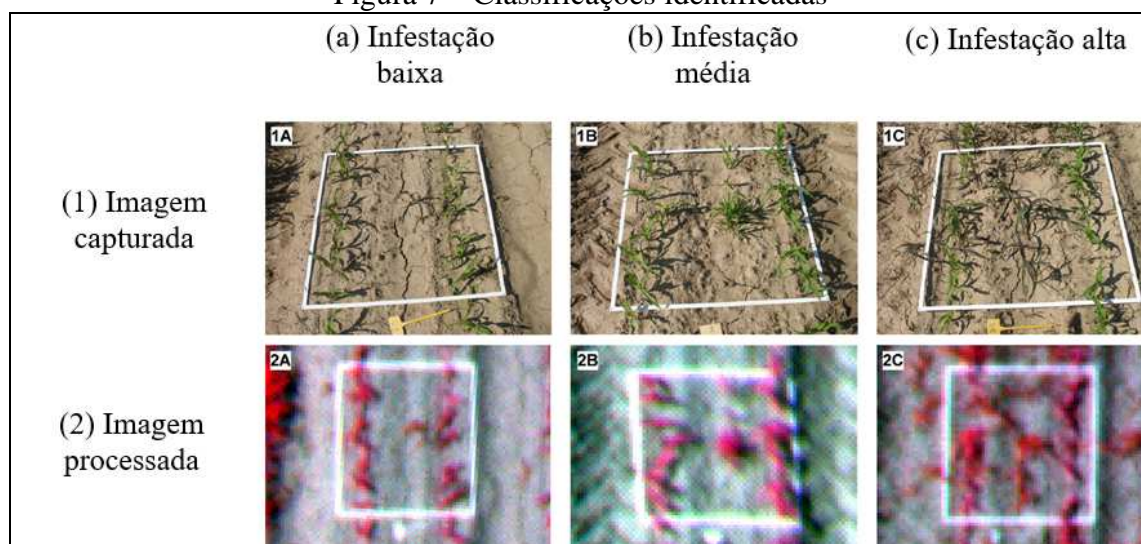
O autor aponta que problemas relacionados ao desempenho da aplicação precisam ser tratados em pesquisas futuras. Como o volume de dados para aplicações na AP tende a ser grande, o desempenho da aplicação é vinculado à capacidade de processamento do servidor, limitações de conexão e o desempenho da máquina do cliente ao desenhar as imagens vetoriais no formato Scalable Vector Graphics (SVG) pelo navegador.

### 2.3.3 Weed mapping in early-season maize fields using object-based analysis of unmanned aerial vehicle (UAV) images

Peña et al. (2013) direcionam seu trabalho para o processamento de imagens capturadas por VANTs para a geração de mapas de pragas no cultivo de milho em estado inicial. O estudo partiu de imagens capturadas em um campo experimental localizado na Espanha, onde foi gerado um mapa atestando a localização e nível de infestação de ervas daninhas.

O procedimento envolveu inicialmente o processamento das imagens a fim de classificar as linhas da cultura, depois a categorização da cultura e das ervas daninhas em diferentes grupos e por fim, a geração de um mapa com as devidas classificações. A partir da Figura 7 é possível identificar as diferentes classificações e as respectivas imagens processadas.

Figura 7 – Classificações identificadas



Fonte: adaptado de Peña et al. (2013).

Para validar o funcionamento do software, Peña et al. (2013) posicionaram algumas placas com marcadores na lavoura para gerar um mapa a partir do Índice de Vegetação por Diferença Normalizada (NDVI). Outros índices também foram validados, porém o NDVI apresentou os resultados mais satisfatórios. O software categorizou corretamente as diferentes áreas do campo de acordo com a infestação de pragas, possibilitando calcular o custo e quantidade de herbicidas necessários para o tratamento das áreas infestadas.



### 3 DESENVOLVIMENTO DO PROTÓTIPO

Este capítulo apresentará todo o ciclo de desenvolvimento do protótipo desenvolvido, contendo os requisitos a serem trabalhados, a especificação e por fim, os resultados obtidos.

#### 3.1 REQUISITOS

No Quadro 1 são relacionados os Requisitos Funcionais (RF) e os Casos de Uso (UC) do protótipo desenvolvido.

Quadro 1 – Matriz de rastreabilidade de RF e UC

Requisitos funcionais	Casos de uso
RF1: permitir o upload de imagens	UC01
RF2: permitir o cadastro de amostras para processamento	UC03 e UC04
RF3: gerar um mapa apontando áreas de interesse na imagem com base nos padrões definidos pelo usuário	UC02
RF4: mensurar os resultados envolvendo a área de recorrência de cada uma das amostras informadas em percentual	UC02

Fonte: elaborado pelo autor.

O Quadro 2 apresenta a relação entre os Requisitos Não-Funcionais (RNF) e os Casos de Uso do presente trabalho.

Quadro 2 – Matriz de rastreabilidade de RNF e UC

Requisitos não-funcionais	Casos de uso
RNF1: ser desenvolvido para a plataforma web a partir do <i>framework</i> JavaServer Faces	UC01, UC02, UC03 e UC04
RNF2: armazenar os dados em uma base de dados NoSQL	UC01, UC02, UC03 e UC04
RNF3: utilizar a tecnologia MATLAB para o processamento de imagens	UC02

Fonte: elaborado pelo autor.

#### 3.2 ESPECIFICAÇÃO

A especificação do protótipo foi representada através de diagramas da Unified Modeling Language (UML), utilizando a ferramenta StarUML. Neste trabalho foram desenvolvidos os diagramas de casos de uso e de atividades. A seguir são apresentadas as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

##### 3.2.1 Técnicas e ferramentas utilizadas

O protótipo foi desenvolvido utilizando a linguagem de programação Java na IDE Eclipse versão Mars e o processamento das imagens com o software MATLAB versão R2015a. As tecnologias utilizadas no desenvolvimento estão listadas abaixo:

- JavaServer Faces 2.2: é uma especificação Java para a construção de interfaces de usuário baseadas em componentes para aplicações web;

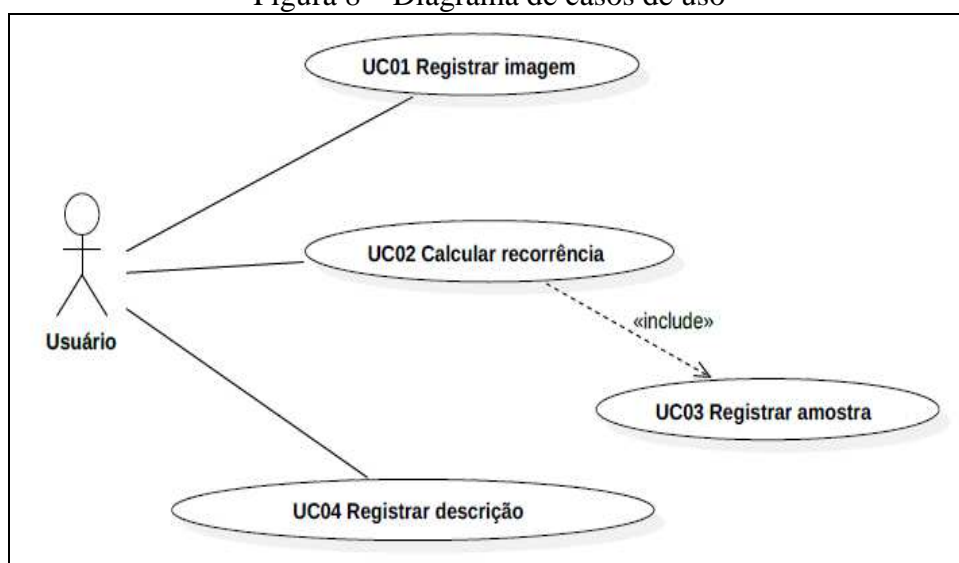
- b) Primefaces 3.5: é um dos principais *frameworks* para a criação de interfaces gráficas para o JSF. O *framework* disponibiliza uma grande quantidade de componentes como campos de formulário, listas, galerias de imagens e gráficos;
- c) MATLAB R2015a: é um software interativo de alta performance voltado para o cálculo numérico. O MATLAB integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em um ambiente fácil de usar, onde problemas e soluções são expressos somente como eles são escritos matematicamente, ao contrário da programação tradicional;
- d) MongoDB 3.2.8: é um banco de dados orientado a documentos de alta performance, open source e schema-free, escrito em C++. Ele é uma mistura entre os repositórios escaláveis baseados em chave/valor e a tradicional riqueza de funcionalidades dos bancos relacionais.
- e) Maven 3: é uma ferramenta de integração de projetos. É responsável por gerenciar dependências, controlar versão de artefatos, gerar informações de produção, garantir a execução de testes, organizar a qualidade do código dentre outras.

Nas próximas seções são detalhados os passos adotados na implementação do protótipo que calcula a recorrência de uma amostra dentro de uma imagem.

### 3.2.2 Diagrama de casos de uso

Nesta seção é apresentado o diagrama de casos de uso, exibido na Figura 8. Identificou-se apenas um ator, denominado *Usuário*, o qual comanda todas as funcionalidades do protótipo.

Figura 8 – Diagrama de casos de uso



Fonte: elaborado pelo autor.

No UC01 - Registrar imagem o protótipo exibe uma janela para que o usuário possa selecionar uma imagem do seu computador. Após o carregamento da imagem, ela é armazenada no banco de dados e uma miniatura fica disponível na lista de imagens. Na lista de imagens, o usuário tem a opção de excluir uma imagem carregada.

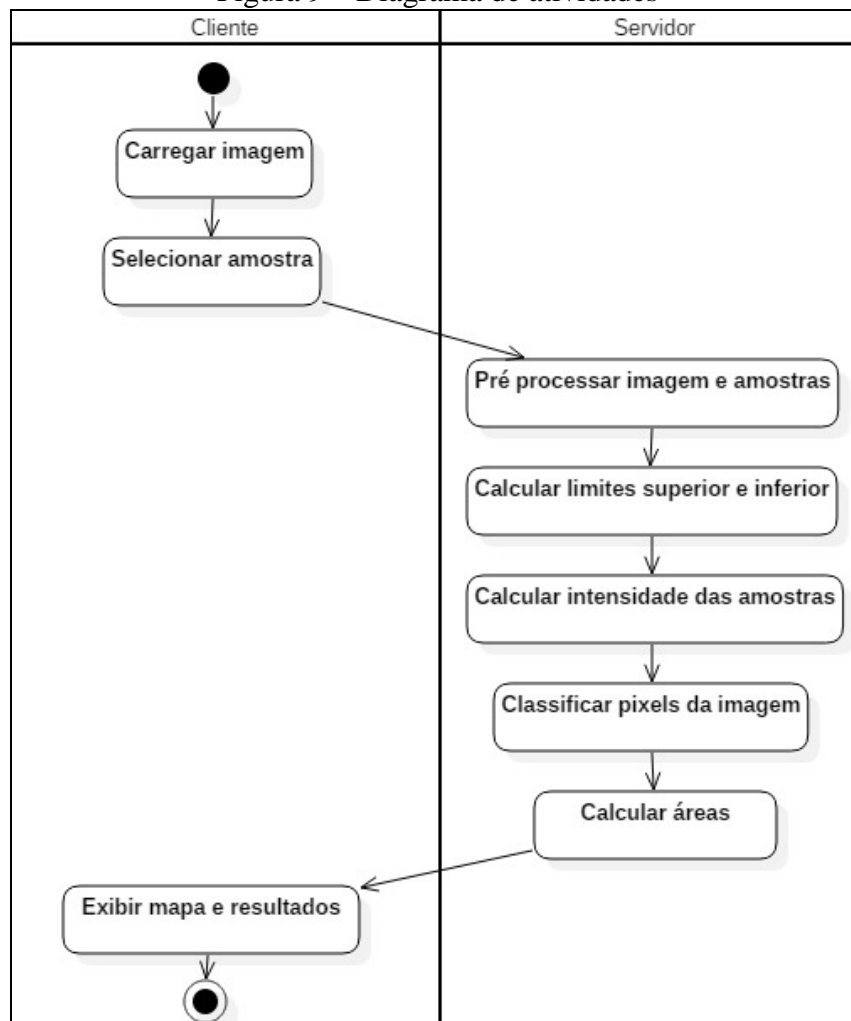
No caso de uso UC02 - Calcular recorrência o usuário pode gerar um mapa com a recorrência de cada uma das amostras. Além do mapa, é apresentada a área de cada uma das regiões identificadas. O caso de uso UC03 - Registrar amostra faz referência à inclusão de novas amostras dentro da imagem. O usuário pode também excluir uma amostra.

Finalmente, o UC04 - Registrar descrição permite a inclusão ou alteração da descrição de uma amostra no banco de dados. A alteração é feita diretamente na tabela de amostras.

### 3.2.3 Fluxo principal do protótipo

O diagrama de atividades demonstra o fluxo dos processos realizados pelo método de identificação de áreas em uma imagem. A Figura 9 apresenta o diagrama com as atividades que o protótipo realiza desde o carregamento da imagem até a geração dos resultados.

Figura 9 – Diagrama de atividades



Fonte: elaborado pelo autor.

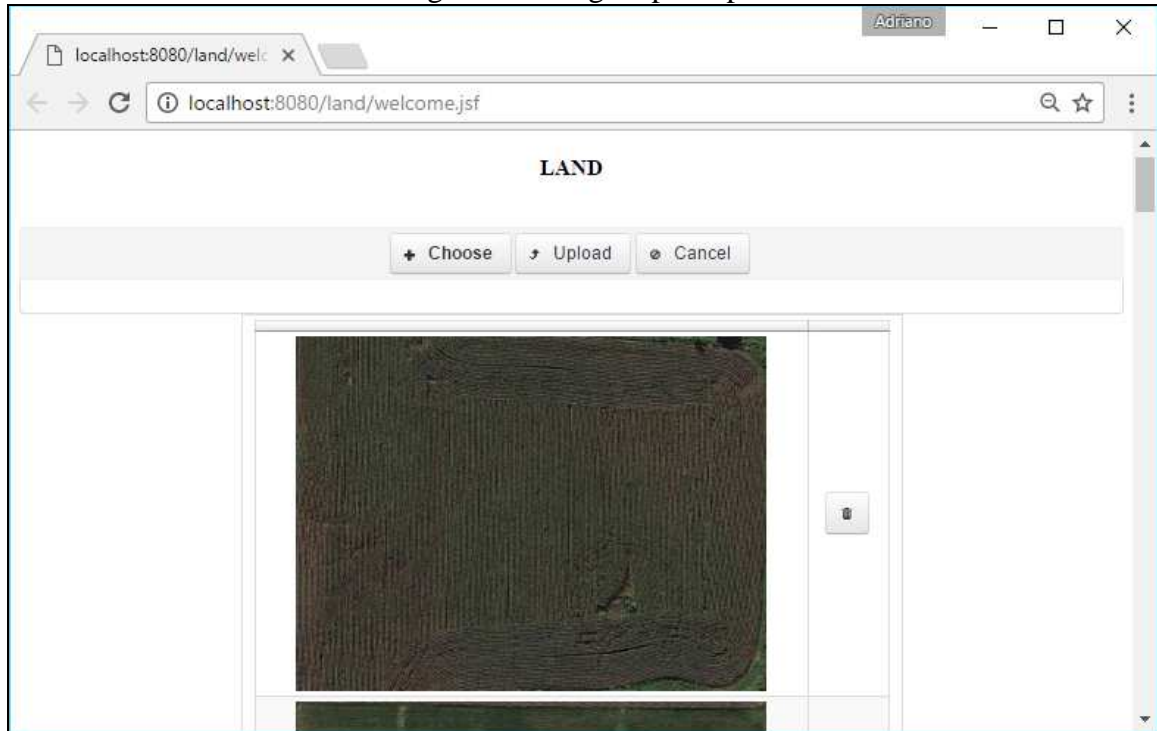
Na Figura 9, observa-se que o protótipo utiliza a arquitetura cliente-servidor. Isto é, as tarefas são distribuídas entre o fornecedor do serviço (o servidor) e o requerente do serviço (o cliente) sendo que ambos se comunicam através de uma rede de computadores ou residem em uma mesma máquina.

As etapas relacionadas na Figura 9 contemplam inicialmente a ação do usuário, que envolvem o upload das imagens aéreas e seleção das respectivas amostras a serem processadas. Depois, no servidor, tem-se início a etapa de processamento das imagens. Um pré-processamento é realizado nas imagens e amostras, logo depois são calculados os limites para classificação dos pixels e por fim, são realizados o cálculo das áreas e a exibição dos resultados.

### 3.2.3.1 Upload de imagens e seleção de amostras

O protótipo desenvolvido contempla uma página principal e uma página de edição. Na página principal é exibida a lista das imagens carregadas, além dos botões para upload e exclusão de imagens. A Figura 10 mostra a tela principal do protótipo.

Figura 10 – Página principal

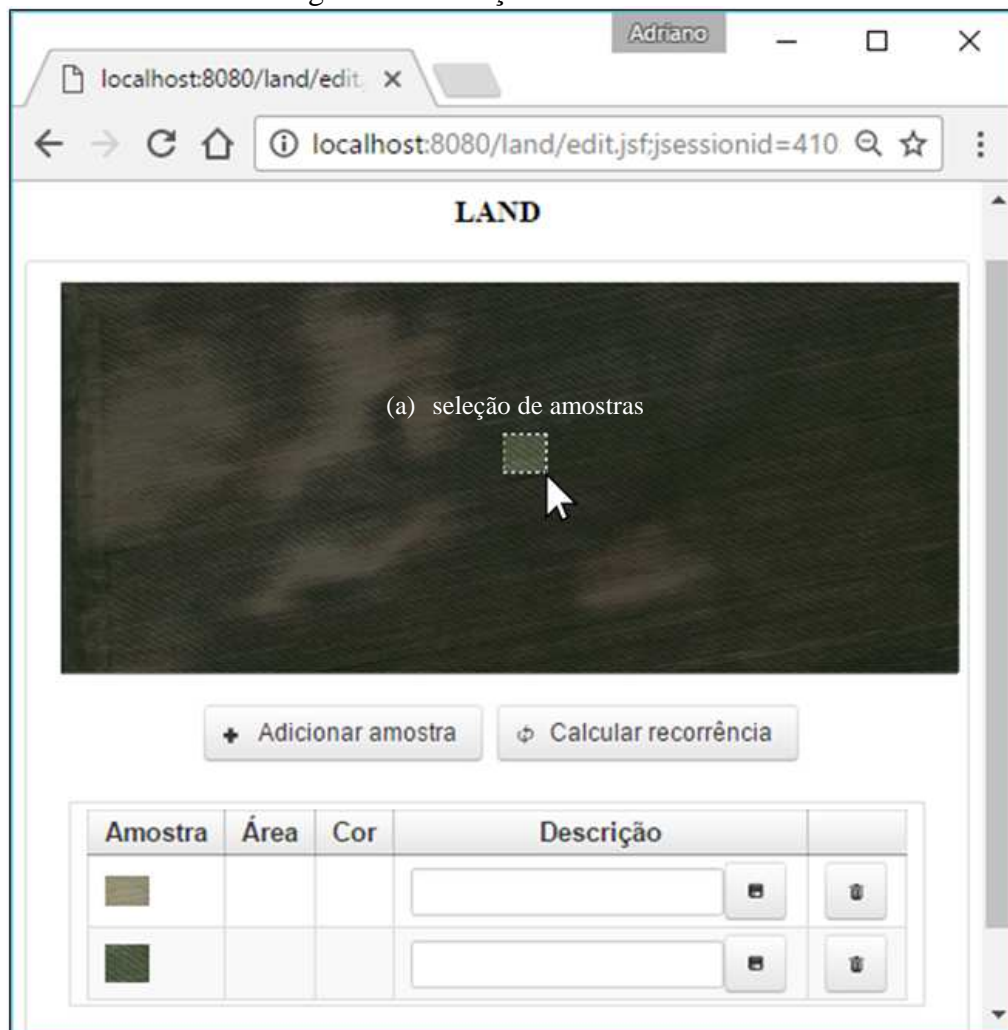


Fonte: elaborado pelo autor.

Para carregar uma nova imagem, o usuário deve clicar em **Choose**. Uma tela de seleção será exibida, na qual o usuário pode selecionar arquivos em sua máquina local. As extensões aceitas pelo protótipo são PNG, GIF, JPG e JPEG. Depois de carregadas, as imagens são exibidas na forma de miniaturas na lista de imagens.

O acesso à página de seleção de imagens é possível a partir do clique sobre uma miniatura da lista de imagens. Nessa página, os componentes incluem a imagem em tamanho original na parte superior, logo abaixo os botões **Adicionar amostra** e **Calcular recorrência** são exibidos e finalmente a tabela de amostras, como apresentado na Figura 11.

Figura 11 – Seleção de uma amostra



Fonte: elaborado pelo autor.

O recurso para seleção de amostras foi implementado a partir da biblioteca PrimeFaces, utilizando o *widget* ImageAreaSelect. Através dela, foi implementada a seleção de segmentos de uma imagem na tela de edição do protótipo, utilizando o comando de clicar e arrastar sobre a imagem exibida.

No item (a) da Figura 11 é apresentado o resultado obtido. O usuário tem a possibilidade de selecionar uma amostra na imagem, a qual é armazenada automaticamente na forma de um arquivo temporário no servidor. Se o usuário desejar, o arquivo temporário criado pode ser armazenado no banco de dados, sendo exibido sempre que a página de edição da respectiva imagem for utilizada.

Ao clicar em *Calcular recorrência*, inicia-se o processamento de imagem que ao final, exibe os resultados. Os valores correspondentes às áreas são atualizados na tabela de imagens e o mapa gerado é exibido logo abaixo da tabela. As próximas seções detalham o método de identificação de áreas e geração de mapas.

### 3.2.3.2 Pré-processamento da imagem e amostras

Visto que o protótipo foi desenvolvido utilizando o *framework* JSF, fez-se necessária a utilização da biblioteca `Matlabcontrol`. Ela possibilita a chamada de uma função externa do software MATLAB a partir de código Java. No Apêndice A é detalhada a utilização das funções da biblioteca `Matlabcontrol`.

Para aperfeiçoar o resultado da classificação, as imagens passaram por um pré-processamento que diminui o ruído no mapa gerado. O filtro aplicado nas imagens foi o `medfilt2`, que realiza a filtragem mediana de uma matriz *A* de duas dimensões. Cada pixel de saída contém o valor médio da área de 3 por 3 pixels em torno do pixel correspondente na imagem de entrada. Esta função é aplicada individualmente em cada um dos três canais RGB das imagens. Para isso, fez-se necessária a conversão da imagem para uma matriz numérica com a fórmula `im2double`. Isso acontece em um laço como apresentado no Quadro 3.

Quadro 3 – Pré-processamento

```

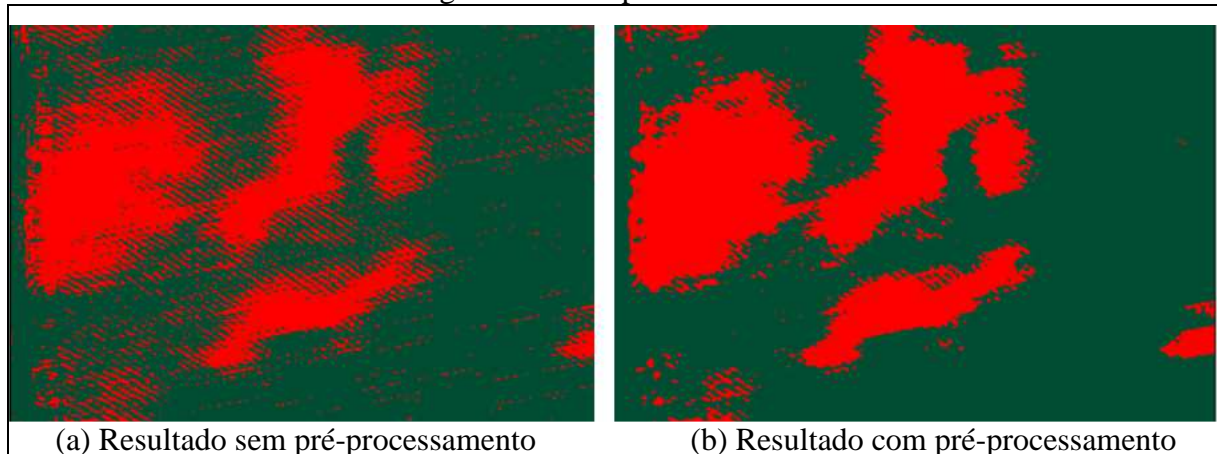
1. % For RGB
2. for z = 1:3
3.     for k=1:nColors
4.         Pattern{k}=im2double(imread([strcat(path, 'matlab\'),
                                           patternFilename{k}])));
5.         %Apply filter for patterns
6.         Pattern{k}(:, :, z) = medfilt2(Pattern{k}(:, :, z));
7.     End
8.     % Apply filter for image
9.     he(:, :, z) = medfilt2(he(:, :, z));
10. End

```

Fonte: elaborado pelo autor.

A Figura 12 apresenta um comparativo dos resultados obtidos com e sem o pré-processamento das imagens, no qual o filtro `medfilt2` foi aplicado.

Figura 12 – Pré-processamento



Fonte: elaborado pelo autor.

Observa-se que o ruído identificado na imagem pré-processada é muito inferior. O pré-processamento gera resultados finais mais precisos, refletindo na qualidade da geração de mapas.

### 3.2.3.3 Cálculo dos limites superior e inferior e a intensidade das amostras

Para cada pixel da imagem analisada, definiu-se uma faixa de intensidade. Esses valores foram utilizados para classificar o pixel em uma ou mais áreas identificadas. Os melhores resultados foram obtidos com uma faixa de 25% abaixo e acima da intensidade do pixel. O Quadro 4 mostra o trecho do código que estabelece o cálculo dos limites inferior e superior.

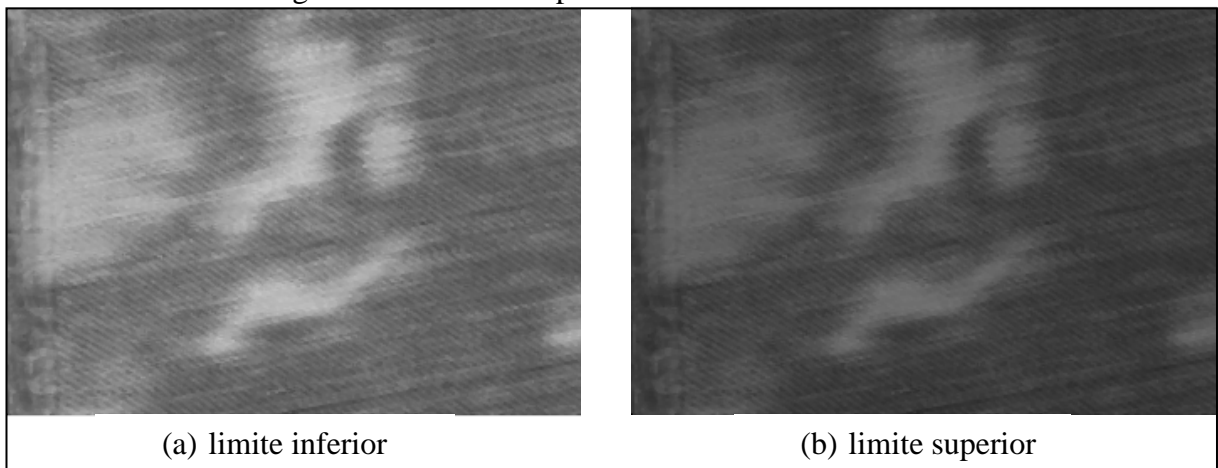
Quadro 4 – Cálculo dos limites inferior e superior

1.	<code>heLowerLimit=(he(:, :, 1)+he(:, :, 2)+he(:, :, 3))/3*0.75;</code>
2.	<code>heHigherLimit=(he(:, :, 1)+he(:, :, 2)+he(:, :, 3))/3*1.25;</code>

Fonte: elaborado pelo autor.

A Figura 13 ilustra um exemplo para as matrizes inferior e superior em escala de cinza.

Figura 13 – Limites superior e inferior de intensidade

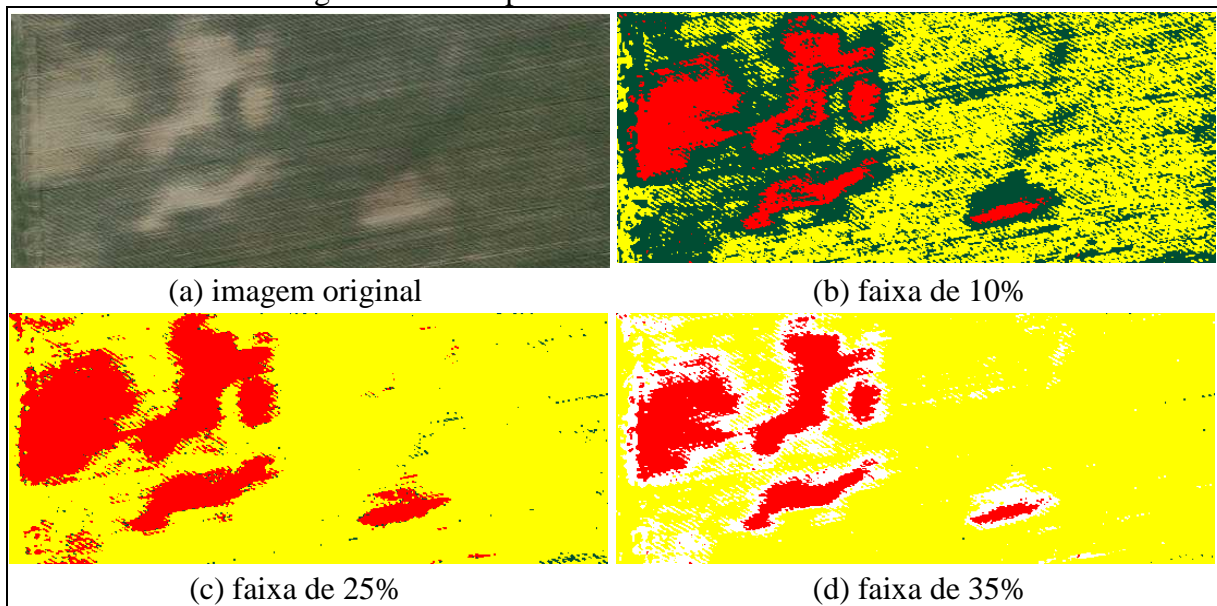


Fonte: elaborado pelo autor.

O valor médio obtido é calculado com base nos três canais de cores RGB para depois calcular-se o limite inferior – 75% do valor original – e superior – 125% do valor original. A faixa de 25% foi definida com base na assertividade da categorização de áreas, como exemplificado no item (c) da Figura 14. Nela, duas amostras foram utilizadas para o cálculo.



Figura 14 – Comparativo das faixas de intensidade



Fonte: elaborado pelo autor.

Para definir a faixa de intensidade mais adequada, as variações analisadas envolveram os valores percentuais inteiros entre 10 e 35% de um em um por cento. Comparando visualmente os itens (b), (c) e (d) da Figura 14 observa-se que no item (c) a definição das duas áreas vermelha e amarela é mais precisa que nas outras variações. No item (b), uma grande área ficou sem categoria (cor verde), enquanto no item (d) uma área foi definida para duas categorias diferentes (cor branca).

Com os limites calculados, o próximo passo foi identificar a intensidade de cada amostra e validar se estas enquadram-se em alguma das regiões da imagem em análise. A intensidade das amostras é calculada através da média aritmética da cor obtida nos três canais de cada uma das amostras, por meio da fórmula `mean` do MATLAB. O cálculo acontece em um laço `for` e é armazenado em um vetor com o número de posições igual ao número de amostras sendo utilizadas no cálculo, conforme atribuição apresentada no Quadro 5.

Quadro 5 – Definição da intensidade das amostras

```
1. average(1, z) = mean(Pattern{z}(:));
```

Fonte: elaborado pelo autor.

A partir dos valores obtidos, tem-se uma intensidade de cor associada a cada amostra. A Figura 15 apresenta a intensidade obtida, em escala de cinza, a partir de uma amostra selecionada pelo usuário.

Figura 15 – Intensidade de uma amostra



Fonte: elaborado pelo autor.

A intensidade apresentada no item (b) da Figura 15 é utilizado na comparação seguinte, com a intensidade de cor de cada pixel, como especificado na próxima seção.

### 3.2.3.4 Classificação dos pixels da imagem e cálculo das áreas

Uma vez definidas as intensidades das amostras, inicia-se o processo de classificação dos pixels. A intensidade de cada pixel é comparada com a intensidade de cada uma das amostras. O trecho de código apresentado no Quadro 6 mostra a implementação, que acontece dentro de um laço percorrendo as amostras ( $z$ ) e os pixels (coordenadas  $x$  e  $y$ ).

Quadro 6 – Definição dos limites de intensidade

```

1. for z = 1:nColors
2.     average(1, z) = mean(Pattern{z}(:));
3.     aux = 0;
4.     for x = 1:nrows
5.         for y = 1:ncols
6.             if average(1, z) > heLowerLimit(x,y) && average(1, z) <
                                   heHigherLimit(x,y)
7.                 if (pixel_labels(x,y) == 0)
8.                     pixel_labels(x,y) = z;
9.                 else
10.                    % Both patterns for one pixel. 5 = white
11.                    pixel_labels(x,y) = 5;
12.                    % Count pixels for area calculation
13.                    aux = aux+1;
14.                end
15.            end
16.        end
17.        area(1, z) = (sum(pixel_labels(:) == z)+aux)*100 / (ncols*nrows);
18.    end

```

Fonte: elaborado pelo autor.

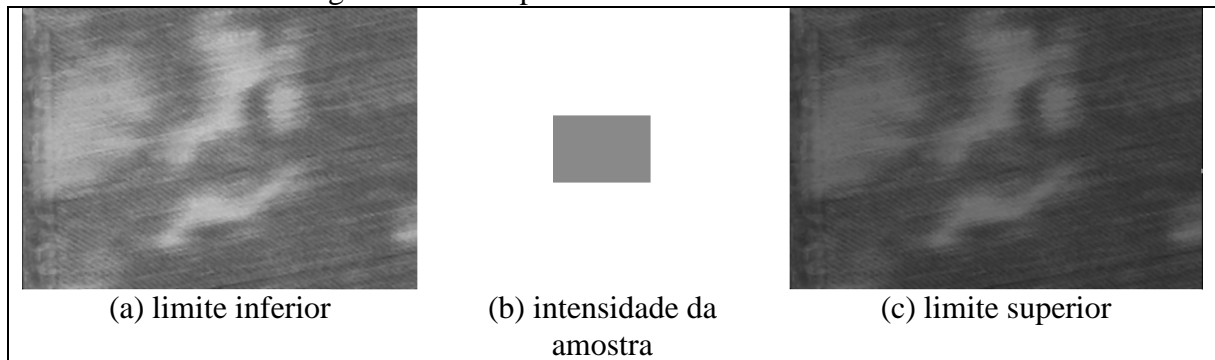
No método de segmentação implementado a definição de cada segmento é feita a partir de um critério pré-definido. Esse critério faz referência às amostras ou fragmentos da imagem informados pelo usuário antes do processamento, referenciado na linha 2 do Quadro 6.

O método de Barbu (2003) fundamentado na seção 2.2 apoiou a definição do algoritmo utilizado no protótipo desenvolvido. Embora no presente estudo não se faça uso do método “crescimento de regiões”, os objetivos da abordagem de Barbu (2003) são condizentes com os objetivos do presente trabalho. Em ambos os trabalhos, as classes (ou critérios para definição cada segmento) são inseridas interativamente pelo usuário. Além disso, a cada *loop* ocorre

atribuição de uma nova região a um segmento, levando em consideração a função classificadora definida.

No Quadro 6 a primeira condição `if` (linha 6) verifica se o valor é compatível com a faixa de intensidade da amostra. As matrizes `heLowerLimit` e `heHigherLimit` representam as escalas de cinza e são ilustradas nos itens (a) e (c) da Figura 16. A matriz `average` representa a intensidade de uma amostra e é ilustrada no item (b) da Figura 16 (também em escala de cinza).

Figura 16 – Comparativo entre limites e amostra



Fonte: elaborado pelo autor.

A matriz `pixel_labels` (mapa final) tem o tamanho da imagem original e é marcada com a valor da respectiva amostra. Se o valor for compatível com mais de uma amostra, o valor 5 (posteriormente definido como cor branca) é marcada no mapa final (linha 11).

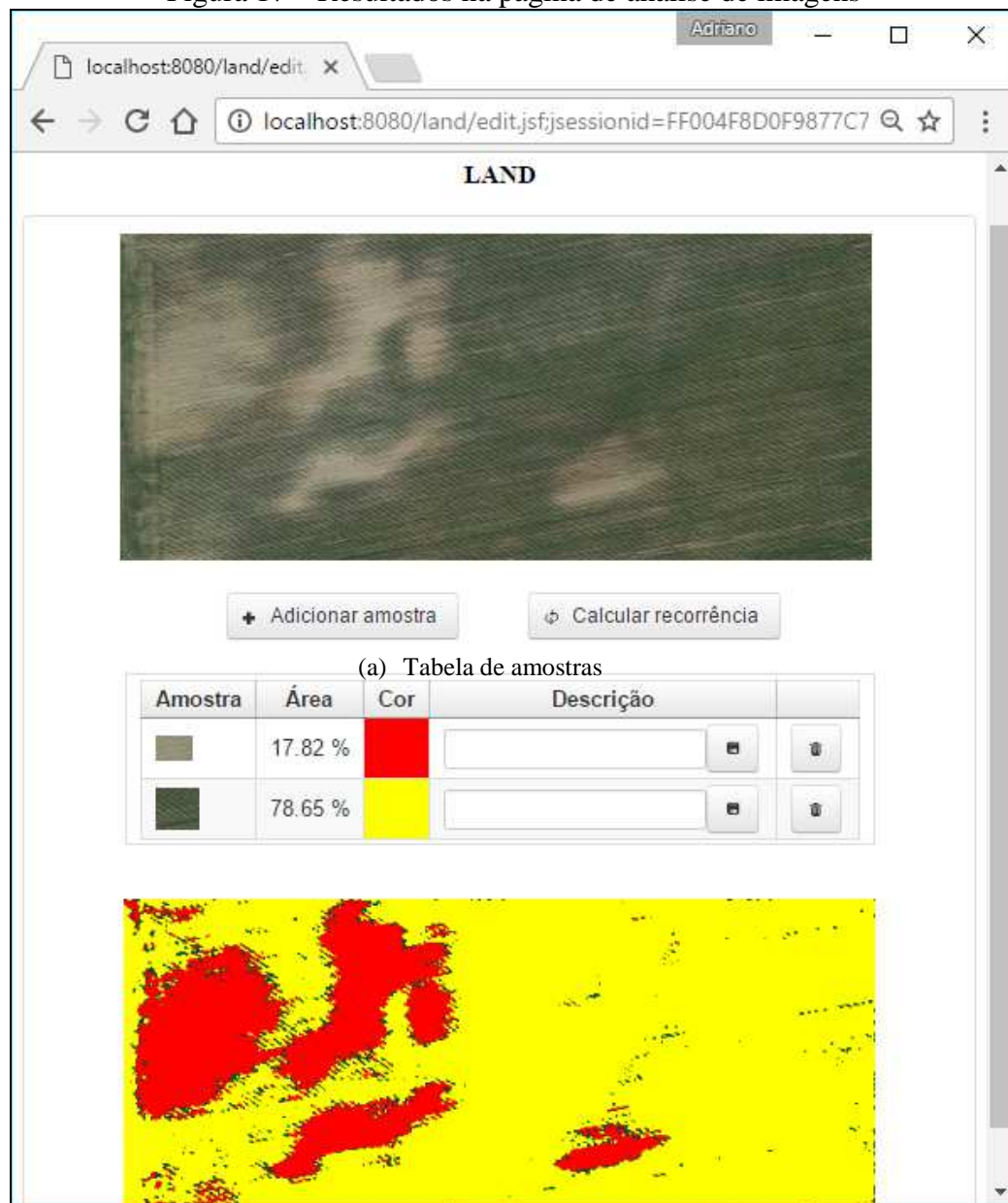
Depois da classificação dos pixels o mapa final está completo. A partir dele é possível obter o percentual para cada área. Para obter o valor em percentual, a quantidade de pixels de cada região é comparada com o número total de pixels.

A relação entre a área calculada no mapa ocorre por meio da coluna `Cor` na Figura 17, apresentada na próxima seção. Ela representa a cor do mapa onde houve recorrência da figura exibida na coluna `Amostra`.

### 3.2.3.5 Exibição do mapa e resultados

Finalmente, o mapa é gerado e salvo em um arquivo temporário. No mapa final cada região é identificada com uma cor diferente, a partir dos valores da matriz `pixel_labels` (0, 1, 2, 3, 4 ou 5). O arquivo temporário é exibido na interface do usuário juntamente com a tabela dos resultados, que exhibe a área que cada uma das amostras foi identificada na imagem original, conforme apresentado na Figura 17.

Figura 17 – Resultados na página de análise de imagens



Fonte: elaborado pelo autor.

O item (a) da Figura 17 representa a tabela de amostras. Nela é possível observar o percentual de recorrência de cada amostra, que corresponde à área total da imagem analisada. O usuário tem a opção de excluir ou adicionar amostras a qualquer momento e gerar um novo mapa com novos dados. Além disso, é possível adicionar uma breve descrição para cada amostra, na coluna *Descrição*.

### 3.3 ANÁLISE DOS RESULTADOS

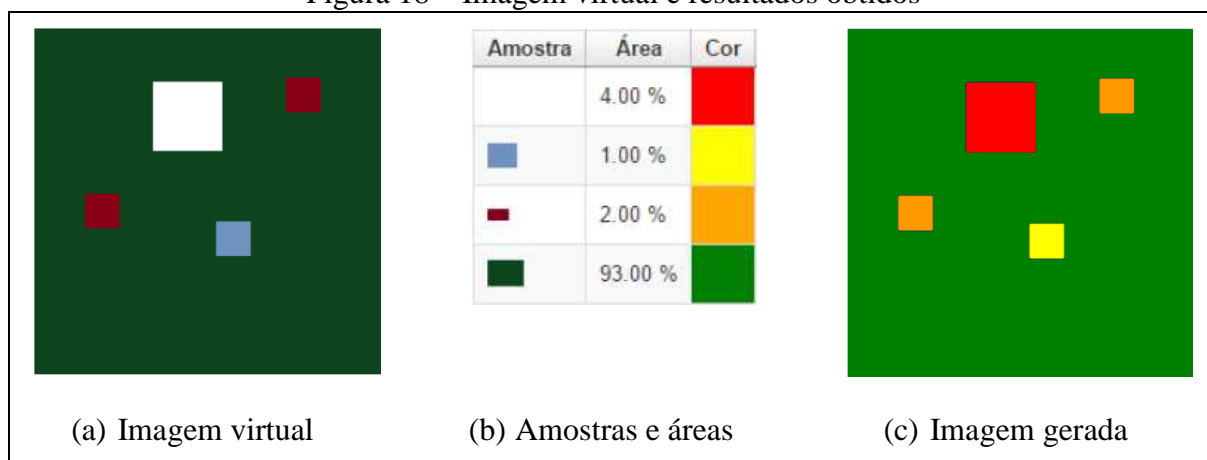
A análise dos resultados obtidos com o protótipo desenvolvido se deu inicialmente por testes com imagens virtuais elaboradas pelo autor. Depois, uma seleção com vinte imagens obtidas a partir do software Google Earth foi definida e utilizada como caso de teste. As

próximas seções apresentam os resultados obtidos e a comparação com os trabalhos correlatos.

### 3.3.1 Testes com imagens virtuais

Primeiramente, a análise dos resultados foi feita a partir de imagens virtuais. O objetivo do teste foi validar a assertividade das áreas calculadas, sendo que as áreas refletem a recorrência das amostras definidas pelo usuário. A Figura 18 apresenta uma imagem virtual elaborada pelo autor e o resultado obtido com a análise do protótipo.

Figura 18 – Imagem virtual e resultados obtidos



Fonte: elaborado pelo autor.

O item (a) da Figura 18 representa uma imagem carregada pelo usuário. O item (b) mostra as amostras selecionadas pelo usuário e as áreas de recorrência de cada uma delas na imagem. Observa-se que no item (c) da Figura 18 é apresentado o mapa gerado, no qual cada área é representada pela respectiva cor do item (b).

Nesse caso de teste, a assertividade do método foi de 100%. Isso é comprovado pela soma dos valores das áreas identificadas (que corresponde a 100% da área total) e pelos percentuais obtidos para cada área. Por exemplo, para uma das amostras o percentual calculado foi de 1%. O valor é compatível com o esperado já que essa área é composta por 2500 pixels de um total de 250000 pixels da imagem virtual. A próxima seção apresenta os resultados obtidos nos testes feitos com imagens aéreas.

### 3.3.2 Testes com imagens aéreas










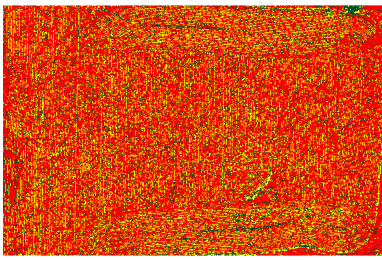




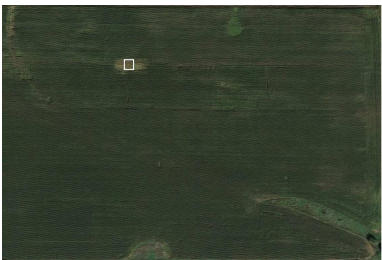




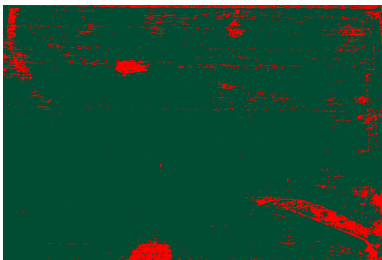


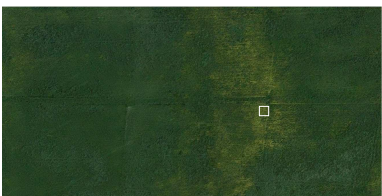




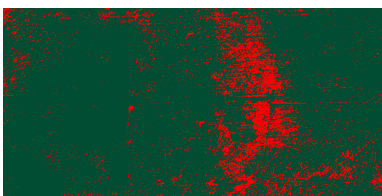







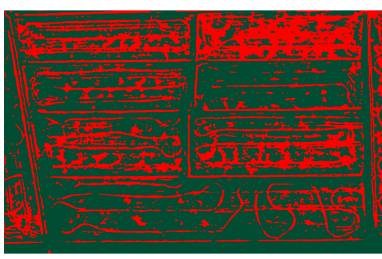











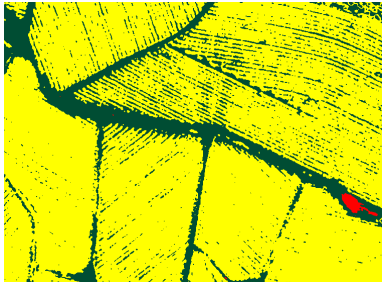




Uma seleção com vinte imagens áreas de campos agrícolas foi gerada pelo autor a partir do software Google Earth. Com esse software, é possível coletar imagens de satélite de paisagens presentes no planeta Terra. A seleção levou em consideração campos agrícolas que



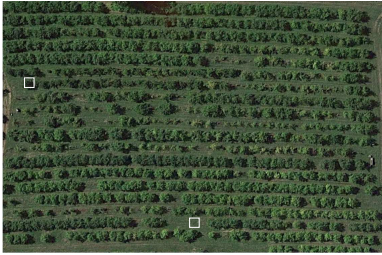








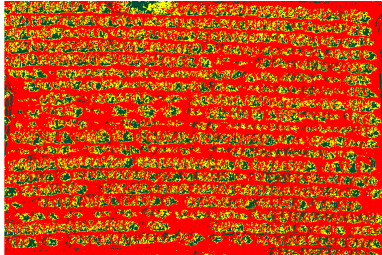













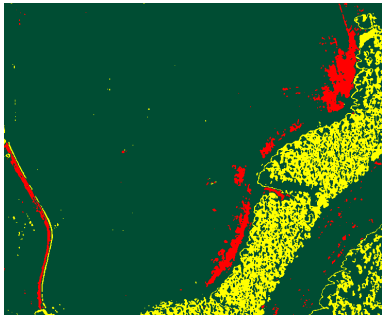













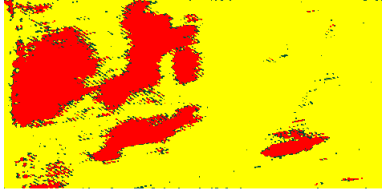









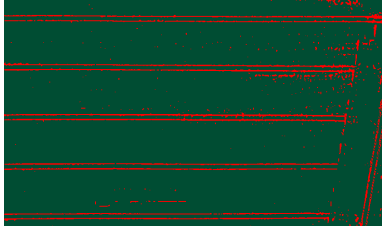


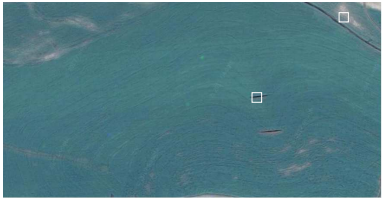








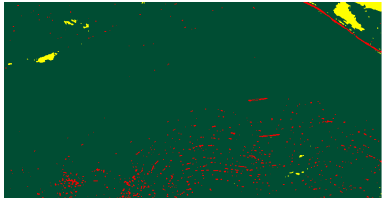




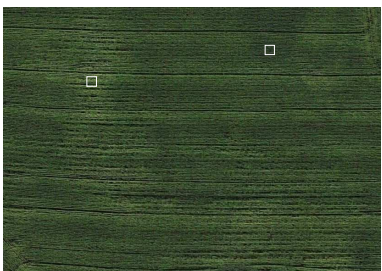








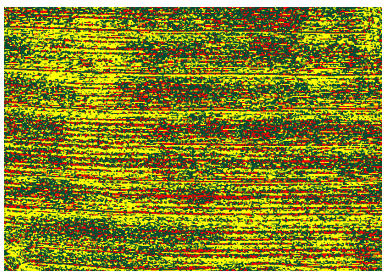




apresentaram variação visual em sua superfície, como por exemplo falhas de plantio, regiões férteis e inférteis, deficiência hídrica, supostas pragas, dentre outros.

O Quadro 7 apresenta as vinte imagens identificadas e os resultados obtidos a partir das amostras selecionadas. O objetivo desse teste foi comparar visualmente as imagens carregadas e os mapas gerados.

Quadro 7 – Resultados alcançados (Continua)










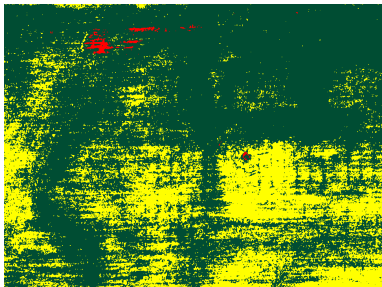




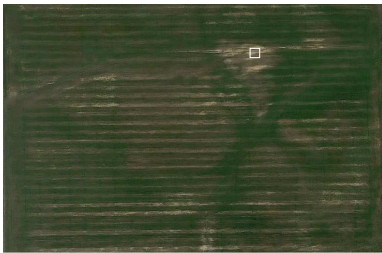




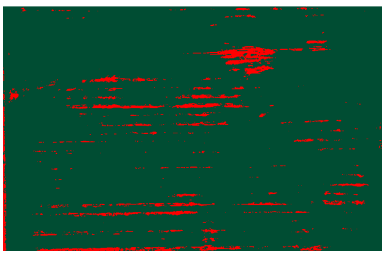















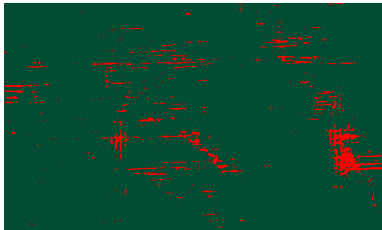







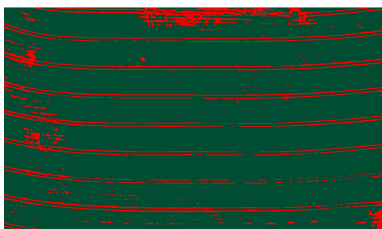


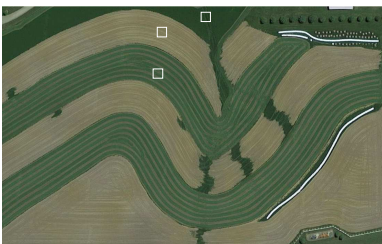



















(a) Imagens aéreas	(b) Amostras e áreas	(c) Mapas gerados									
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>61.53 %</td><td></td></tr> <tr> <td></td><td>22.47 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		61.53 %			22.47 %		
Amostra	Área	Cor									
	61.53 %										
	22.47 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>8.21 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		8.21 %					
Amostra	Área	Cor									
	8.21 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>11.02 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		11.02 %					
Amostra	Área	Cor									
	11.02 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>31.05 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		31.05 %					
Amostra	Área	Cor									
	31.05 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>0.26 %</td><td></td></tr> <tr> <td></td><td>82.04 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		0.26 %			82.04 %		
Amostra	Área	Cor									
	0.26 %										
	82.04 %										

Quadro 7 – Resultados alcançados (Continuação)

	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>59.85 %</td><td></td></tr> <tr> <td></td><td>20.10 %</td><td></td></tr> </table>	Amostra	Área	Cor		59.85 %			20.10 %		
Amostra	Área	Cor									
	59.85 %										
	20.10 %										
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>2.98 %</td><td></td></tr> <tr> <td></td><td>10.34 %</td><td></td></tr> </table>	Amostra	Área	Cor		2.98 %			10.34 %		
Amostra	Área	Cor									
	2.98 %										
	10.34 %										
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>17.82 %</td><td></td></tr> <tr> <td></td><td>78.65 %</td><td></td></tr> </table>	Amostra	Área	Cor		17.82 %			78.65 %		
Amostra	Área	Cor									
	17.82 %										
	78.65 %										
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>6.48 %</td><td></td></tr> </table>	Amostra	Área	Cor		6.48 %					
Amostra	Área	Cor									
	6.48 %										
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>1.31 %</td><td></td></tr> <tr> <td></td><td>0.95 %</td><td></td></tr> </table>	Amostra	Área	Cor		1.31 %			0.95 %		
Amostra	Área	Cor									
	1.31 %										
	0.95 %										
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>16.13 %</td><td></td></tr> <tr> <td></td><td>34.82 %</td><td></td></tr> </table>	Amostra	Área	Cor		16.13 %			34.82 %		
Amostra	Área	Cor									
	16.13 %										
	34.82 %										



Quadro 7 – Resultados alcançados (Continuação)

	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>0.38 %</td><td></td></tr> <tr> <td></td><td>27.76 %</td><td></td></tr> </table>	Amostra	Área	Cor		0.38 %			27.76 %					
Amostra	Área	Cor												
	0.38 %													
	27.76 %													
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>4.96 %</td><td></td></tr> </table>	Amostra	Área	Cor		4.96 %								
Amostra	Área	Cor												
	4.96 %													
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>0.71 %</td><td></td></tr> </table>	Amostra	Área	Cor		0.71 %								
Amostra	Área	Cor												
	0.71 %													
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>3.41 %</td><td></td></tr> </table>	Amostra	Área	Cor		3.41 %								
Amostra	Área	Cor												
	3.41 %													
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>9.72 %</td><td></td></tr> </table>	Amostra	Área	Cor		9.72 %								
Amostra	Área	Cor												
	9.72 %													
	<table> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> <tr> <td></td><td>41.13 %</td><td></td></tr> <tr> <td></td><td>47.98 %</td><td></td></tr> <tr> <td></td><td>14.66 %</td><td></td></tr> </table>	Amostra	Área	Cor		41.13 %			47.98 %			14.66 %		
Amostra	Área	Cor												
	41.13 %													
	47.98 %													
	14.66 %													



Quadro 7 – Resultados alcançados (Conclusão)

	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>21.11 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		21.11 %					
Amostra	Área	Cor									
	21.11 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>15.31 %</td><td></td></tr> <tr> <td></td><td>80.85 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		15.31 %			80.85 %		
Amostra	Área	Cor									
	15.31 %										
	80.85 %										
	<table border="1"> <thead> <tr> <th>Amostra</th><th>Área</th><th>Cor</th></tr> </thead> <tbody> <tr> <td></td><td>12.85 %</td><td></td></tr> </tbody> </table>	Amostra	Área	Cor		12.85 %					
Amostra	Área	Cor									
	12.85 %										

Fonte: elaborado pelo autor.

Os pontos marcados em branco na coluna (a) do Quadro 7 identificam a região na qual cada uma das amostras foi selecionada pelo usuário. Analisando visualmente os mapas gerados, percebe-se a assertividade dos resultados em todas as imagens, já que se tem uma visão nítida e bem definida de cada uma das áreas identificadas. Os casos com resultados insatisfatórios caracterizam em sua maioria imagens e/ou amostras de má qualidade, e são detalhadas nas limitações do protótipo apresentadas na seção 4.1.

Dessa forma, associados às áreas calculadas (exibidas na coluna (b) do Quadro 7), esses mapas têm como objetivo auxiliar o gestor da produção agrícola a analisar a variabilidade espacial da lavoura e assim suportar o planejamento das ações a serem tomadas para impulsionar a rentabilidade da produção. A próxima seção compara os resultados obtidos no presente estudo e os trabalhos correlatos.

### 3.3.3 Comparativo dos resultados com os trabalhos correlatos

No Quadro 8 é apresentado um comparativo entre os trabalhos correlatos e o protótipo desenvolvido. As linhas representam as características e as colunas cada um dos trabalhos.

Quadro 8 – Comparativo entre os trabalhos correlatos e o trabalho atual

Características \ Correlatos	López et al. (2015)	Ribeiro Júnior (2007)	Peña et al. (2013)	Graebin (2017)
controla o dispositivo para captura das imagens	X		X	
utiliza processamento digital de imagens	X		X	X
permite a visualização de um mapa com resultados	X	X	X	X
análise a partir de amostras definidas pelo usuário				X

Fonte: elaborado pelo autor.

Pode-se observar a partir do Quadro 8 que o escopo dos trabalhos de López et al. (2015) e Peña et al. (2013) está direcionado à captura e o processamento digital de imagens. Porém, as aplicações abordam problemas muito específicos da AP. Eles são aplicáveis para determinadas culturas e problemas de produtividade, mas não possibilitam ao usuário muita flexibilidade se este desejar identificar ou mensurar áreas específicas dentro do campo produtivo.

O trabalho de Ribeiro Júnior (2007) trata de um *framework* para a construção de aplicações no âmbito da AP. Este, individualmente, não permite ao usuário que funcionalidades da AP sejam utilizadas, isto só é possível a partir de um subproduto deste *framework*. As aplicações construídas não preveem o processamento de imagens, mas sim o processamento de dados textuais ou numéricos extraídos do campo, definindo aspectos como produtividade, umidade, presença de pragas, dentre outros.

Diante das limitações identificadas, o protótipo de software desenvolvido contemplou aspectos diferentes dos tratados nos trabalhos relacionados. No presente trabalho, não é levado em consideração o meio de captura das imagens aéreas. Nele, utiliza-se o processamento digital de imagens para a geração de mapas com resultados e o usuário tem autonomia para a definição de áreas de interesse dentro da imagem – ele informa uma ou mais amostras como padrão a ser processado. A partir disso, a variabilidade espacial de uma plantação pode ser calculada e comparada com inúmeros parâmetros, obtidos pelos padrões informados pelo usuário.

## 4 CONCLUSÕES

Este trabalho compreende o desenvolvimento de um protótipo de software para a análise de imagens aéreas de alta resolução. O propósito das imagens e dados processados é a avaliação da variabilidade espacial de uma plantação.

O desenvolvimento do protótipo foi realizado utilizando a IDE Eclipse versão Mars e a linguagem de programação Java, mais especificamente o *framework* JSF. Para o processamento de imagens a tecnologia MATLAB foi empregada. As tecnologias utilizadas facilitaram o desenvolvimento e ofereceram recursos que possibilitaram o produto final atender os requisitos especificados.

O primeiro objetivo do trabalho era realizar a identificação de padrões a partir de amostras informadas pelo usuário dentro de uma imagem aérea de uma plantação. O método adotado pelo autor fez uso da intensidade de uma amostra a partir da cor obtida pelos canais RGB de cada um de seus pixels. A assertividade na definição destes dados foi fundamental e refletiu diretamente nos demais objetivos.

Percebe-se que a representação visual dos resultados gerados pelo protótipo desenvolvido satisfaz o segundo objetivo proposto. Este envolveu a geração de um mapa identificando áreas de recorrência de cada uma das amostras informadas pelo usuário. Em cada mapa gerado, as regiões são demarcadas com sua respectiva cor, associada à amostra previamente definida.

Os resultados envolvendo a área e a proporção de cada uma das áreas em questão refletem o terceiro objetivo estabelecido. Os dados são exibidos ao usuário com valores percentuais, juntamente com o mapa gerado.

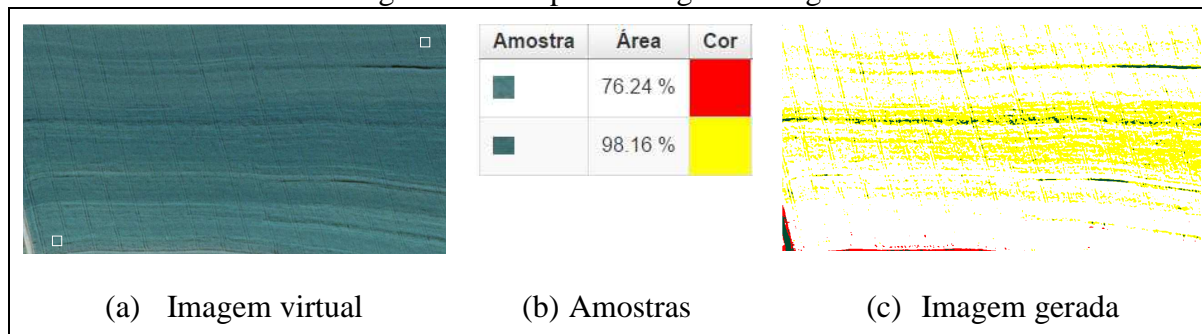
De forma geral, os resultados se mostraram satisfatórios. Onde, conclui-se que o presente trabalho, aliado a uma ferramenta que permita a captura de imagens aéreas pode ser útil para produtores que ainda fazem uso de métodos convencionais de produção, sem qualquer tipo de automatização ou processo que facilite a análise. Partindo deste protótipo, ferramentas voltadas para a análise computacional de imagens aéreas no âmbito da AP podem ser desenvolvidas permitindo uma tomada de decisão mais precisa e automatizada.

### 4.1 LIMITAÇÕES

As limitações do protótipo desenvolvido estão relacionadas à assertividade na análise de imagens com amostras muito semelhantes. Quando o usuário seleciona amostras parecidas, elas podem ser associadas à uma única categoria. Nesse caso, a região em divergência é

marcada com a cor branca no mapa. Essa área é acrescida no percentual de todas as regiões identificadas. Na Figura 19 é exibido um exemplo de imagem com região divergente.

Figura 19 – Mapa com região divergente



Fonte: elaborado pelo autor.

Percebe-se que no mapa gerado (item (c) da Figura 19) há uma vasta área marcada em branco. Essa região apresentou recorrência das duas amostras selecionadas e, dependendo dos objetivos da análise, pode representar resultados insatisfatórios. Para uma representação visual mais apurada, faz-se necessária uma análise visual independente para cada amostra.

## 4.2 EXTENSÕES

Como sugestões de extensão para o presente trabalho propõe-se:

- implantar o georreferenciamento das regiões identificadas nos mapas gerados;
- implementar a captura das imagens de entrada por VANTs;
- desenvolver técnicas para categorização automática das imagens trabalhadas como pragas, falhas de plantio, áreas com deficiência hídrica, dentre outros;
- desenvolver recursos para a diferenciação de imagens de uma mesma plantação em diferentes estágios da produção;
- diferenciar cada região nas áreas marcadas como sobreposição nos mapas gerados;
- implementar um módulo para estimar a produtividade a partir de dados coletados na plantação nas áreas identificadas pelo protótipo;
- diferenciar cada região nas áreas marcadas como sobreposição nos mapas gerados.

## REFERÊNCIAS

- ANTONOPOULOU, Eleni et al. Web and mobile technologies in a prototype DSS for major field crops. **Computers and electronics in agriculture**, [s.l.], v. 70, n. 2, p.292-301, mar. 2010. Elsevier BV. Disponível em: <<http://dx.doi.org/10.1016/j.compag.2009.07.024>>. Acesso em: 12 set. 2016.
- BARBU, Tudor. A pattern recognition approach to image segmentation. **Proceedings of the Romanian academy**, [s.l.], v. 4, n. 2, mar. 2003. Disponível em: <[http://www.acad.ro/sectii2002/proceedings/doc2-2003/10\\_barbu.pdf](http://www.acad.ro/sectii2002/proceedings/doc2-2003/10_barbu.pdf)>. Acesso em: 01 jun. 2017.
- BERNARDI, Alberto Carlos de Campos et al. **Agricultura de precisão: resultados de um novo olhar**. Brasília-DF: Embrapa, 2014. 596 p.
- BIANCHI, Marcelo F. **Extração de características de imagens de faces humanas através de wavelets, PCA e IMPCA**. 2006. 135 f. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2006.
- DOMINGUES JUNIOR, Gilberto; INAMASU, Ricardo. Proposta de uma metodologia para utilização da agricultura de precisão de acordo com a estratégia adotada pelas empresas do agronegócio brasileiro. In: CONGRESSO BRASILEIRO DE AGRICULTURA DE PRECISÃO, 7., 2010, Ribeirão Preto/SP.
- GOMES, Otavio F. M. **Microscopia co-localizada: Novas Possibilidades na Caracterização de Minérios**. 2007. 105 f. Tese (Doutorado) - Curso de Engenharia Metalúrgica e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007.
- GONZALEZ, Rafael C.; WOODS, Richard E. **Digital image processing**. 3. ed. Upper Saddle River-NJ, EUA: Prentice Hall, 2007. 976 p.
- JENSEN, Allan L. et al. Pl@nteInfo® — a web-based system for personalised decision support in crop management. **Computers and electronics in agriculture**, [s.l.], v. 25, n. 3, p.271-293, fev. 2000. Disponível em: <[http://dx.doi.org/10.1016/s0168-1699\(99\)00074-5](http://dx.doi.org/10.1016/s0168-1699(99)00074-5)>. Acesso em: 08 set. 2016.
- LAMPARELLI, Rubens A. C.; ROCHA, Jansle V.; BORGHI, Elaine. **Geoprocessamento e agricultura de precisão: Fundamentos e Aplicações**. Guaíba-RS: Livraria e Editora Agropecuária, 2001. 118p.
- LÓPEZ, Andrés F. J. et al. Crops diagnosis using digital image processing and precision agriculture technologies. **Inge cuc**, [s.l.], v. 11, n. 1, p.63-71, jan. 2015. Disponível em: <<http://dx.doi.org/10.17981/ingecuc.11.1.2015.06>>. Acesso em: 10 set. 2016.
- MURAKAMI, Edson. **Uma infra-estrutura de desenvolvimento de sistemas de informação orientados a serviços distribuídos para agricultura de precisão**. 2006. 186p. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo. 2006.
- NATIONAL RESEARCH COUNCIL. **Precision agriculture in the 21st century: Geospatial and Information Technologies in Crop Management**. Washington DC, EUA: National Academy Press, 1997, 168 p.
- PEÑA, José M. et al. Weed mapping in early-season maize fields using object-based analysis of unmanned aerial vehicle (UAV) images. **Plos One**, [s.l.], v. 8, n. 10, p.1-11, out. 2013. Disponível em: <<http://dx.doi.org/10.1371/journal.pone.0077151>>. Acesso em: 02 set. 2016.

RIBEIRO JÚNIOR, Luiz C. M. **Uma arquitetura de software para sistemas espaço-temporais baseados na web para agricultura de precisão**. 2007. Tese (Doutorado em Sistemas Digitais) - Escola Politécnica, Universidade de São Paulo, São Paulo, 2007.

SARAIVA, Antonio M. **Tecnologia da informação na agricultura de precisão e biodiversidade**: Estudos e proposta de Utilização de Web Services para Desenvolvimento e Integração de Sistemas. 2003. Tese (Livre Docência) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2003. 187p.

SUGIURA, Ryo et al. Field information system using an agricultural helicopter towards precision farming. **Advanced Intelligent Mechatronics**, [s.l.], p.1073-1078, jul. 2003. Disponível em: <<http://dx.doi.org/10.1109/aim.2003.1225491>>. Acesso em: 15 set. 2016.

THEODORIDIS, Sergios; KOUTROUMBAS, Konstatinos. **Pattern recognition**. 4. ed. Burlington-MA, EUA: Academic Press, 2008. 984 p.

VANDERLEEST, Zach et al. **Choosing the right imagery**: Best Management Practices for Color, NIR, and NDVI Imagery. Disponível em:

<<http://crops.extension.iastate.edu/cropnews/2016/05/choosing-right-imagery-best-management-practices-color-nir-and-ndvi-imagery>>. Acesso em: 31 out. 2016.

VIBHUTE, Anup; BODHE, Shrikant K. Applications of image processing in agriculture: a survey. **International Journal Of Computer Applications**, [s.l.], v. 52, n. 2, p.34-40, ago. 2012. Disponível em: <<http://dx.doi.org>

## APÊNDICE A – CONFIGURAÇÃO DA API MATLABCONTROL

`Matlabcontrol` é uma API Java que permite a chamada de funções MATLAB em sessões locais ou remotas. As chamadas acontecem no código Java e as funções `.m` retornam objetos que podem ser manipulados diretamente na aplicação Java. É possível incluir a biblioteca em um projeto a partir da configuração de uma dependência Maven no arquivo `pom.xml`, a qual é apresentada no Quadro 9

Quadro 9 – Dependência Maven

```

1. <!-- https://mvnrepository.com/artifact/org.n52.matlab/matlab-control
   -->
2. <dependency>
3.   <groupId>org.n52.matlab</groupId>
4.   <artifactId>matlab-control</artifactId>
5.   <version>5.0.0</version>
6. </dependency>

```

Fonte: elaborado pelo autor.

A função utilizada no presente estudo foi a `returningFeval`, a qual chama uma função MATLAB com o nome `runAnalysis.m`, retornando o resultado. Os argumentos para a função podem ser fornecidos como `args`, mas não são necessários se a função não precisar de argumentos.

Os argumentos da função são convertidos a argumentos equivalentes no MATLAB. Isso significa que uma `String` será convertida para uma matriz MATLAB `char`, não para um nome de variável. Para que o resultado dessa função seja retornado, o número de argumentos a serem devolvidos deve ser especificado. Isso é equivalente no MATLAB ao número de variáveis colocadas no lado esquerdo de uma expressão. Por exemplo, no MATLAB, a função `inmem` pode ser usada com valores de retorno de 1, 2 ou 3 – cada um resultando em um comportamento diferente, como apresentado no Quadro 10.

Quadro 10 – Notação das funções `.m`

```

1. M = inmem;
2. [M, X] = inmem;
3. [M, X, J] = inmem;

```

Fonte: elaborado pelo autor.

A matriz com os objetos retornados terá o comprimento de cada argumento de retorno colocado na posição de matriz correspondente. Se a função não for capaz de retornar o número de argumentos especificados, uma exceção será lançada. Na linha 8 do Quadro 11, é apresentada a chamada da função `runAnalysis.m` utilizada no presente estudo.

Quadro 11 – Biblioteca matlabcontrol

```
1. if (proxy == null) {  
2.     MatlabProxyFactoryOptions options = new MatlabProxyFactoryOptions  
3.     .Builder().setUsePreviouslyControlledSession(true).build();  
4.     MatlabProxyFactory factory = new MatlabProxyFactory(options);  
5.     proxy = factory.getProxy();  
6. }  
7. proxy.eval("addpath(' " + scriptPath + "')");  
8. // Save area  
9. Object[] data = proxy.returningFeval("runAnalysis", 1, args);
```

Fonte: elaborado pelo autor.

Observa-se na linha 8 do Quadro 11 que o retorno é armazenado em um *array* denominado *data*. A partir desse objeto é extraído o retorno da função *.m* – assim é possível manipular os resultados finais calculados.