

Plano de Trabalho: Identificação de Falhas em Linha de Plantio de Cana Usando Visão Computacional

Daniel Ortega de Carvalho^{a,b}, Hemerson Pistori (Orientador)^{b,c}

^aRA: 170088, Curso: Engenharia de computação, Semestre: 3 semestre

^bUniversidade Católica Dom Bosco, Campo Grande, MS, Brazil

^cFederal University of Mato Grosso do Sul, Campo Grande, MS, Brazil

Abstract

Este plano de trabalho propõe ajustar e aprimorar técnicas de agricultura de precisão voltada ao monitoramento do plantio da cana-de-açúcar por meio de visão computacional. Um software em Python será criado, fundamentando-se nos algoritmos baseados na transformada de Hough, para detectar falhas em linhas de plantio de cana, por meio de imagens aéreas obtidas por VANTs. Desse modo, o trabalho de acompanhamento do crescimento da plantação de cana será automatizado, preciso e praticamente imediato.

Keywords: Visão Computacional, Plantio em linha, Cana-de-açúcar, Transformada de Hough

1. Introdução

O processo de acompanhamento do crescimento da cana-de-açúcar é essencial para o controle de seu cultivo e sua manutenção [1]. Com isso, o uso de tecnologias baseadas em análises de dados por meio de programas de computadores tornou-se uma necessidade na agropecuária, especialmente as de grandes extensões, cuja quantidade e complexidade de dados e informações inviabiliza o uso de ferramentas não automatizadas para interpretá-los e classificá-los [2].

A implementação da automação da informação em processos agrícolas mostrou-se cada vez mais necessária nos estados com economia estruturada principalmente no agronegócio [3]. Em Mato Grosso do Sul essa situação não é diferente: o controle e manutenção, e até a supervisão das plantações é uma tarefa extremamente complexa e extensa, processo no qual se mostra necessário o uso de ferramentas computacionais para eficientemente obter e manipular grandes quantidades de dados.

Diante deste quadro, o recente estudo aplicado das tecnologias de visão computacional na agricultura apresentou soluções muito eficiente nessa área. Isso possibilita analisar massivas quantidades de dados em um período viável de tempo quando se utiliza algoritmos de identificação automática de certos padrões desejáveis. Como consequência disso, aumenta-se a eficiência no meio agrícola, com o controle supervisionado tanto da produtividade quanto dos danos ou falhas em plantações [4].

Em diversas situações, o plantio em linha da cana-de-açúcar apresenta, por diferentes motivos, falhas que geral-

mente são padronizadas. O processo manual de identificação com precisão dessas falhas é algo que demanda muito tempo e trabalho devido à grande quantidade de informações e dados a serem classificados. É nesse momento que a ideia da criação de um software baseado em visão computacional entra para solucionar esse problema, visando automatizar todo esse processo.

Normalmente, para este tipo de problema, usa-se métodos de análise de imagens, em que fotos aéreas de um campo agrícola capturadas por VANTs são as entrada para um programa de computador [5]. Este modelo de software tem o objetivo de informar onde há falhas e erros no plantio e, de modo geral, identificar visualmente as áreas afetadas. Além disso, também se pode criar uma base de dados que envolva todas essas fotos do campo, gerando gráficos matemáticos e visuais para que todas as variáveis envolvidas no problema sejam unificadas, facilitando a compreensão de algo muito maior e mais complexo, para que desse modo, decisões mais precisas e apropriadas sejam tomadas visando o aumento da produtividade agrícola da cana-de-açúcar.

A ideia geral do projeto é desenvolver um software na linguagem de programação Python [6], baseando-se em bibliotecas de manipulação de imagens, convolução de matrizes, e aprendizado de máquina. Posteriormente a isso, a partir dos métodos e parâmetros criados, aplicar algoritmos de visão computacional baseados em adaptações da transformada de Hough para o reconhecimento de padrões em imagens aéreas de linha retas ou curvadas de plantio da cana-de-açúcar, com o objetivo principal de identificar e classificar as falhas nesses plantios.

Email addresses: daniel.ortega.carvalho@gmail.com (Daniel Ortega de Carvalho), pistori@ucdb.br (Hemerson Pistori (Orientador))

2. Objetivos

2.1. Geral

Desenvolver um software baseado em métodos de visão computacional utilizando fotos aéreas de plantios, com o objetivo de apontar e classificar locais de falhas nas plantações em linha de cana-de-açúcar. Subsequente a isso, criar gráficos e tabelas baseados nesses dados obtidos para melhorar a compreensão do motivo dessas falhas, para que desse modo se possa otimizar os métodos do plantio a fim de evitá-las. Sendo assim, dar auxílio tecnológico à agricultura estadual e desenvolvê-la para esta atender eficientemente não só toda a região envolta a ela, como também atender aos padrões de demanda e qualidade internacionais da agroexportação.

2.2. Específicos

A fim de atingir o objetivo geral apresentado na Seção 2.1, definiu-se os seguintes objetivos específicos:

- Criar um banco de dados de imagens obtidas através da captura de fotos aéreas por meio de VANTs.
- Desenvolver um software em Python capaz de identificar e anotar as falhas nas plantações de cana-de-açúcar em linha.
- Analisar o desempenho dos algoritmos, otimizando o que for necessário, e criar um software final partir disso.

3. Revisão de literatura

Nesta seção serão apresentados os métodos em que este trabalho científico se baseará para criar o software de identificação de falhas em linha de plantio de cana usando visão computacional

3.1. Superpixel

A aplicação de visão computacional na agricultura envolve diferentes técnicas de processamentos de dados e classificação de imagens. Muitos trabalhos científicos usam métodos de identificação de padrões nas plantações para logo em seguida identificar problemas envolvendo crescimento de plantas parasitas indesejáveis, como a erva-daninha, no plantio de monocultura [7]. Para isto, tem-se mostrado eficiente o uso de segmentação de imagens através do algoritmo de Superpixel, cujo objetivo é criar um mosaico que fraciona a imagem e separa suas frações por características em comum, desse modo há uma diminuição artificial na resolução da imagem sem que esta perca sua qualidade original. Esse método facilita o trabalho na visão computacional, pois classifica os pontos mais notáveis de uma imagem, além de diminuir o tempo de processamento de máquina por criar um pixel artificial muito maior do que o original.

3.2. Regressão polinomial de grau variável

Como nem sempre as linhas de plantação da cana-de-açúcar apresentam padrões lineares contínuos (Figura 1), o uso restrito de algoritmos de regressão polinomial de primeiro grau (regressão linear) não é recomendável, pois poderiam gráficos não correspondentes com as reais linhas de plantio. Uma das alternativas para solucionar isso seria adicionar o método de algoritmos baseados em regressão polinomial de grau variável. Aplicado à visão computacional, esses algoritmos mostraram-se eficiente ao permitir acompanhar tendências de curvaturas ao longo da linha de plantio da cana. Estudos anteriores mostram que a eficiência desse algoritmo, junto ao de regressão linear, apresenta uma eficiência alta (acima de 90%) para criar imagens com uma linha curvada média que passa pelos pixels notáveis e desejados nas linhas de plantações de milho, viabilizando a sua aplicação na identificação de linhas de plantio cuja planta se destaca em relação ao meio devido as suas discrepâncias de cores, que é também o caso da cana-de-açúcar [8].

3.3. Convolução de matrizes

As imagens digitais não são nada mais do que matrizes com 2 informações: sua posição e sua cor. A informação da sua posição é o próprio índice da matriz, e a sua cor é representada por 3 valores, com 8 bits cada (quando usado o padrão de cor RGB), o "R" (vermelho), "G" (verde) e o "B" (azul). Logo, como são matrizes com valores numéricos, existem procedimentos que usam um método matemático chamado de convolução para criar filtros em imagens. Esse método consiste basicamente em usar uma segunda matriz, que contém um filtro personalizável, fazendo-a deslocar em superposição sobre a matriz principal, que pode ser a imagem, aplicando operações lineares. Como consequência, a imagem é alterada conforme o tipo do filtro utilizado. Neste projeto, serão utilizados alguns tipos filtros para facilitar a separação das áreas de interesse da imagem, que neste caso seriam apenas os terrenos com plantações de cana.

3.4. Transformada de Hough / Espaço de Hough

e Um método bastante eficiente para detectar formas é a transformada de Hough, desenvolvida por Paul Hough em 1962, que trabalha em cima do chamado Espaço de Hough. Para uma dada equação da reta $y = mx + b$, o objetivo desse método é atravessar uma reta por todos os pontos da imagem, rotacionando-a por completo em cada ponto, plotando um gráfico de cada coeficiente angular "m" e de cada constante "b" de cada ponto. Em cada rotação, é acumulado o valor de sobreposições da reta e de outros pontos da imagem a fim de evidenciar no espaço cartesiano de Hough os locais da imagem onde há padrões lineares. O algoritmo da transformada de Hough é muito eficiente e já mostrou se eficiente em sistemas de plantio em linha, como mostrados em [7] e [9]. Outros métodos

baseados na transformada de Hough também serão usados, como a Transformada de Hough Probabilística, cuja vantagem é a maior velocidade de execução comparada ao método tradicional, sem perder a sua precisão de maneira significativa [10].

4. Materiais e Métodos

A seguir, serão apresentados detalhadamente os procedimentos para atingir os objetivos da Seção 2.2.

4.1. Banco de imagens

Será usado um banco de dados com imagens de plantio de cana, cedido por colaboradores do grupo Inovisão¹. Além disso, tem-se como objetivo secundário capturar diversas fotos aéreas (Figura 1) por meio de VANTs, de áreas de plantio em linha de cana-de-açúcar para gerar um novo banco de imagens. Diante disso, serão estudados os melhores e mais eficientes métodos de fotografia aérea para futuramente facilitar o trabalho de detecção das falhas envolvidas na plantação, baseados nos seguintes aspectos:

1. Resolução das fotos
2. Altura da captura de imagem;
3. Estabilidade de voo do VANT
4. Horário do dia para obtenção das imagens

Também serão analisadas as as imagens capturadas em situações não ideais de fotografia, como fotos de baixa resolução e VANTS com menor estabilidade de voo. Com isso, o objetivo é de que VANTs e câmeras mais baratos possam ser usados para detectar as falhas na linha de plantio, o que barateia o custo da operação e abre a possibilidade do pequeno e médio agricultor usufruir dessa tecnologia de maneira fácil e financeiramente viável.



Figure 1: Exemplo do formato da foto aérea de uma área de plantio capturada por um VANT, onde se mostra as diversas linhas de plantações junto às estradas e às vegetações nativas.

4.2. Criação do software em Python

A partir das imagens salvas em um banco de dados, iniciar a execução de métodos de identificações de pontos notáveis na plantação, neste caso, evidenciar os pontos da imagem que têm mais evidência nos tons de verde (cor predominante da cana-de-açúcar), e assim criar uma nova imagem com apenas duas cores: branco para as canas-de-açúcar e preto para os demais objetos do cenário, como o terreno, estradas, áreas alagadas, etc.

Para esse método se usará um filtro de tons de verde, de acordo com o padrão RGB, seguindo o algoritmo que compara os valores de vermelho, verde e azul: Se o valor de verde for maior que o valor do vermelho e o valor do azul, o pixel é pintado de branco, caso contrário, o pixel é pintado de preto. Junto a esse algoritmo, serão aplicados diversos outros filtro que determinam ponderadamente o peso do valor das diferentes cores com tons esverdeados, o que cria possibilidade de trabalhar com fotos com diferentes tonalidades de iluminação ou de cor da cana-de-açúcar.

Outros métodos de espaço de cores também serão usados a fim de comparar seus resultados, e identificar qual se encaixa melhor em cada tipo de vegetação ou terreno. Serão explorados os espaços de cores HSV/HSB e o HSL, pois ambos oferecem uma classificação e manipulação bastante intuitiva à percepção humana do espectro das cores.

Para aplicar os algoritmos de espaços de cores às fotos obtidas do plantio da cana, será usada a biblioteca OpenCV [11], na linguagem de programação Python 3. O código do filtro de cores será testado e adaptado até o momento em que ele detecte com precisão a tonalidade de cor da cana-de-açúcar.

Em seguida, tem-se o objetivo de criar algoritmos para destacarem as bordas de todo o terreno onde se plantou a cana e eliminar tudo que esteja fora dessa área, como estradas, rios, florestas, casas, entre outras coisas. A ideia é usar a imagem aérea e identificar onde está somente a área de plantação, para que assim o algoritmo de detecção trabalhe apenas sobre a área de plantio, pois caso sejam usados dados de imagem que não remetem restritamente à área do plantio, acontecerão erros de identificação do que é uma linha de plantação e do que não é uma linha de plantação.

Para atingir esses objetivos serão usados os algoritmos de Convolução de imagens e de Superpixel, ambos presentes na biblioteca OpenCV do Python 3. Para a convolução, serão usados filtros de destaque de borda, o que facilitarão a identificação do terreno onde há a área de plantio em relação ao meio. Após isso, o método de segmentação de imagens Superpixel será utilizado para separar somente esse terreno onde há a plantação e eliminar dos cálculos de visão computacional as demais áreas externas.

A partir das imagens dos plantios isoladas pelos algoritmos de superpixel, será desenvolvido um método, baseado na transformada de Hough, que separa a imagem em partes com resoluções menores, e dentro dessas partes serão colocadas linhas rasterizas com inclinações aleatórias

¹Website do grupo Inovisão: <http://inovisao.weebly.com/>

em pontos aleatórios. Em seguida, para cada pixel dessas linhas, será avaliado se ele sobrepôs algum pixel da cana-de-açúcar presente na imagem. No final de tudo isso, será levantado uma estatística que selecionará as retas com maiores sobreposição de pixels, identificando a inclinação padrão delas, o que consequentemente mostrará o sentido local das linhas de plantio. Nota-se que pelo fato desse método separar a imagem em áreas menores, ele poderá ser usado em plantações com linhas de plantios curvadas, pois ao ponto de vista de uma subárea da imagem, a curvatura do plantio se apresentará de maneira quase imperceptível, o que pode ser aproximado a uma reta colinear à linha de plantação.

Após desenvolvido o algoritmo citado acima, novamente a biblioteca usada para será o OpenCV, no qual já se tem diversas ferramentas e algoritmos prontos e de alto desempenho para o objetivo desse trabalho científico. Feito isso, será ajustada a parte gráfica e visual da marcação das linhas de plantio, com a indicação de cores para mostrar onde há falhas e onde não há falhas nas imagens de plantações da cana-de-açúcar.

Em seguida, haverá um teste em diferentes modelos de plantações em linha de canas-de-açúcar:

- quanto ao modelo de plantação: serão analisados os tipos de linhas de plantio quanto a sua linearidade ou curvatura, observando se o algoritmo conseguiu identificar corretamente as equações polinomiais que correspondam ao formato espacial dessas linhas com uma precisão mínima que será definida posteriormente.
- quanto às cores: o algoritmo, baseado nos espaços de cores RGB,HSV/HSB e HSL, será testado em plantações com diferentes padrões de cores presentes nas imagens aéreas obtidas, tais padrões sendo influenciados pela espécie da cana, época sazonal, fatores de iluminação (posição relativa do sol às plantações), modelo da câmera do VANT e o tempo atmosférico.
- quanto ao tipo do terreno: os diversos objetos e cenários presentes nas imagens serão analisados e descartados para não prejudicarem a identificação das linhas de plantio. O algoritmo será testado para conseguir identificar as plantações meio ao solo, estradas, rios, casas, animais, vegetações nativas e entre outras coisas, como é mostrado na Figura ??.

4.3. Otimização e criação do software final

A análise de desempenho em testes de visão computacional é essencial para a viabilidade do programa. No contexto da agricultura de precisão, imagens aéreas de plantio precisam ter resoluções altas para não haver interpretação equivocada de dados. Diante disto, interpretar um dado agrícola com visão computacional é algo que demanda um muito processamento de máquina, o que reforça a necessidade dos códigos de programação serem altamente

otimizados para que seu rendimento seja eficiente e tenha rapidez na execução.

Códigos escritos em linguagens interpretadas, como o Python, têm uma alta produtividade, porém são geralmente lentos quando trabalham com manipulação de matrizes com milhões de índices, pois o interpretador limita o desempenho dessa linguagem, como foi mostrado em [12]. Com o objetivo de manter a alta produtividade que o Python oferece e sem perder desempenho, será usada a biblioteca Numba [13], que transforma funções do Python em códigos mais otimizados para linguagem de máquina, fazendo-os aproximarem da velocidade de execução das linguagens C/C++.

Cada parte do algoritmo será analisada individualmente, para ser otimizada e adaptada para funcionar sobre os padrões da biblioteca Numba. Serão feitos testes de desempenho da velocidade média dos algoritmos, direcionados à execução tanto em CPU quanto para GPU, para definir qual dos dois métodos têm o melhor desempenho em cada etapa do algoritmo.

Definido toda a parte de testes de desempenho, com o modelo final escolhido, o software final estará pronto para ser usado. Para facilitar seu uso ao público, será criada uma interface gráfica intuitiva na qual o usuário insere a imagem, ou um conjunto de imagens, da plantio em linha de cana-de-açúcar. O software irá criar as novas imagens datadas e já classificadas que identifique as falhas nas fotos do plantio.

Sucintamente, serão realizadas as seguintes atividades:

1. Estudo de artigos sobre as plantação de cana-de-açúcar em linha no Brasil, de maneira mais específica ao estado de Mato Grosso do Sul
2. Estudo dos seguintes algoritmos: convolução, Superpixel, transformada de Hough
3. Realização de coleta das imagens aéreas de plantio de cana-de-açúcar
4. Adaptar a implementação da transformada de Hough para identificar linhas de plantio de cana
5. Testar o desempenho de todos os algoritmos e, posteriormente, do software final
6. Concluir a escrita do artigo científico reportando os resultados alcançados

A tabela 1 apresenta o cronograma bimestral para as atividades propostas.

Table 1: Cronograma Bimestral de Execução das Atividades Propostas

Ativ.	B1	B2	B3	B4	B5	B6
1	X	X				
2	X	X	X			
3		X	X			
4			X	X		
5				X	X	
6					X	X

References

- [1] Dennis J Casley and Denis A Lury. *Monitoring and evaluation of agriculture and rural development projects*. Number 338.1 C37. World Bank Washington DC, 1982.
- [2] Konstantinos G Liakos, Patrizia Busato, Dimitrios Moshou, Simon Pearson, and Dionysis Bochtis. Machine learning in agriculture: A review. *Sensors*, 18(8):2674, 2018.
- [3] Joonyoung Lee, ShinHo Kim, SaeBom Lee, HyeonJin Choi, and JaiJin Jung. A study on the necessity and construction plan of the internet of things platform for smart agriculture. *Journal of Korea multimedia society*, 17(11):1313–1324, 2014.
- [4] Diego Inácio Patrício and Rafael Rieder. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and electronics in agriculture*, 153:69–81, 2018.
- [5] Leonardo Felipe Maldaner and José Paulo Molin. Data processing within rows for sugarcane yield mapping. *Scientia Agricola*, 77(5), 2020.
- [6] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.
- [7] M Dian Bah, Adel Hafiane, and Raphaël Canals. Deep learning with unsupervised data labeling for weeds detection on uav images. *arXiv preprint arXiv:1805.12395*, 2018.
- [8] Iván García-Santillán, Martín Montalvo Martínez, José Guerrero, and Gonzalo Pajares. Automatic detection of curved and straight crop rows from images in maize fields. *Biosystems Engineering*, 156:61–79, 04 2017.
- [9] Ronghua Ji and Lijun Qi. Crop-row detection algorithm based on random hough transformation. *Mathematical and Computer Modelling*, 54(3-4):1016–1020, 2011.
- [10] Nahum Kiryati, Yuval Eldar, and Alfred M Bruckstein. A probabilistic hough transform. *Pattern recognition*, 24(4):303–316, 1991.
- [11] G. Bradski. The OpenCV Library. *Dr. Dobb’s Journal of Software Tools*, 2000.
- [12] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pages 1–6, 2015.
- [13] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, LLVM ’15, New York, NY, USA, 2015. Association for Computing Machinery.