UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Extração e aprendizado de características espectrais em imagens de sensoriamento remoto de baixa altitude

Paulo Victor de Souza Prado

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Paulo Victor de Souza Prado

Extração e aprendizado de características espectrais em imagens de sensoriamento remoto de baixa altitude

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Moacir Antonelli Ponti

USP – São Carlos
Abril de 2021

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

P896e Prado, Paulo Victor de Souza
Extração e aprendizado de características
espectrais em imagens de sensoriamento remoto de
baixa altitude / Paulo Victor de Souza Prado;
orientador Moacir Antonelli Ponti. -- São Carlos,
2021.
77 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2021.

1. Aprendizado de características. 2.
Processamento de imagens. 3. Sensoriamento remoto.
I. Ponti, Moacir Antonelli, orient. II. Título.

Paulo Victor de Souza Prado

Extraction and learning of spectral-features in low altitude
remote sensing images

Dissertation submitted to the Institute of Mathematics
and Computer Sciences – ICMC-USP – in
accordance with the requirements of the Computer
and Mathematical Sciences Graduate Program, for
the degree of Master in Science. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Prof. Dr. Moacir Antonelli Ponti

USP – São Carlos
April 2021

AGRADECIMENTOS

Primeiramente gostaria de agradecer ao meu orientador Prof. Dr. Moacir Antonelli Ponti que ao longo de todos estes anos sempre esteve a disposição para me ajudar e orientar nos estudos e desenvolvimentos da pesquisa. Também gostaria de agradecer à minha família, amigos e colegas de trabalho que me incentivaram a concluir o projeto, em especial Jean Vitor de Paulo que ajudou a revisar minha pesquisa.

RESUMO

PRADO, P. V. S. **Extração e aprendizado de características espectrais em imagens de sensoriamento remoto de baixa altitude.** 2021. 74 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

O presente trabalho tem como objetivo aplicar estratégias de pré-, pós-processamento e técnicas de aprendizado de máquina, para obter características de imagens capturadas por dispositivos de baixo custo em contextos agrícolas, de forma a melhorar detecção de cobertura vegetal. Os conhecimentos adquiridos a partir destas características podem ser usados diretamente nas práticas de agricultura de precisão; com a captação de dados feita através de dispositivos de baixo custo a implantação do sistema pode atingir diferentes categorias de agricultores de diversos países, principalmente os de países em desenvolvimento. Para a análise das técnicas de processamento de imagem e de aprendizado de máquina, utilizaremos bases de dados de imagens referentes a plantações, em um caso mais específico, plantações de feijão. É descrito um método não supervisionado, capaz de aprender a representação destes dados, sendo aptos a reconstituir sua entrada e utilizar como índices de vegetação ou ainda criar canais novo. Este modelo é conhecido como autoencoder. Os resultados obtidos com o modelo foram comparados com métodos clássicos da literatura que são capazes de estimar vegetação apenas com análises da composição RGB da imagem de entrada, chamados de Índices de Vegetação. Apesar dos índices de vegetação demonstrarem grande relevância para detecção de cobertura vegetal, o uso de auto-supervisão para reconstrução de canais de cor mostrou que é possível obter melhorias através da análise do erro da reconstrução.

Palavras-chave: Aprendizado de características, Processamento de imagens, Sensoriamento remoto.

ABSTRACT

PRADO, P. V. S. **Extraction and learning of spectral-features in low altitude remote sensing images**. 2021. 74 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

The present work proposes the application of pre- post-processing and machine learning strategies to extract features from images captured by low cost devices in agricultural contexts, in order to improve green coverage detection. The knowledge acquired from these features can be used directly in precision agriculture practices. Due to the fact of the data is captured by low cost devices, this system application can reach different categories of farmers from several countries, especially those from developing countries. For the analysis of the image processing and machine learning techniques, we will use databases referring to plantations, in a more specific case, bean plantations. An unsupervised method is described, capable of learning the representation of this data, being able to reconstruct its input and use it as vegetation indexes or create new channels. This model is known as autoencoder. The results obtained with the model were compared with classic methods in the literature that are able to estimate vegetation with only the analysis of the RGB components of the input image, called Vegetation Indexes. Although the vegetation indexes demonstrate great relevance for the coverage detection, the use of self-supervised for the reconstruction of color channels has shown that its possible to obtain improvements through the analysis of the reconstruction error.

Keywords: Feature learning, Image processing, Remote sensing.

LISTA DE ILUSTRAÇÕES

Figura 1 – Ciclo de Produção	26
Figura 2 – Respresentações de diferentes espectros	29
Figura 3 – Representação autoencoder para imagens	33
Figura 4 – Representação otimização com Gradiente Descendente	35
Figura 5 – Arquitetura U-Net	36
Figura 6 – Bloco residual com 2 componentes	36
Figura 7 – Arquitetura rede para reconstrução de canais com entrada contendo apenas 2 canais Red, Green, Blue (Vermelho, Verde, Azul) (RGB)	37
Figura 8 – Validação cruzada - <i>k-fold</i>	38
Figura 9 – Amostras base de dados calibrada	41
Figura 10 – Amostras base de dados não calibrada	42
Figura 11 – <i>Pipeline</i> principal para detecção de cobertura vegetal e passos para melhora nos resultados: setas laranjas indicam pré-processamento da imagens, setas azuis indicam a fusão dos índices de vegetação, e setas verdes indicam a fusão das classificações.	45
Figura 12 – Combinação redes para predição de canais	47
Figura 13 – RGB Receiver Operating Characteristic (Característica de Operação do Re- ceptor) (ROC) plot	50
Figura 14 – Vegetation Index (Índice de Vegetação) (VI) ROC plot	51
Figura 15 – Curva ROC para os algoritmos de pré-processamento	52
Figura 16 – Boxplot da acurácia e AUC para a base de dados 1: Coluna 1 refere-se às imagens sem pré-processamento, e a 2 às imagens suavizadas	54
Figura 17 – Boxplot da acurácia e AUC para a base de dados 2: Coluna 1 refere-se às imagens sem pré-processamento, e a 2 às imagens suavizadas	54
Figura 18 – <i>Loss</i> para treinamento de autoencoder <i>fully connected</i> com 3 camadas e código de tamanho 256	56
Figura 19 – Resultados rede <i>Fully Connected</i>	56
Figura 20 – Curvas ROC para arquitetura de auto-supervisão com otimizador SGD	60
Figura 21 – Curvas ROC para arquitetura de auto-supervisão para base de dados 1	61
Figura 22 – Resultados arquitetura auto-supervisão para base de dados 2	62
Figura 23 – Resultados arquitetura auto-supervisão para base de dados 1	63

LISTA DE ALGORITMOS

Algoritmo 1 – Algoritmo para votação da maioria - <i>Late Fusion</i>	43
--	----

LISTA DE TABELAS

Tabela 1 – Índices de vegetação utilizados nesse estudo	41
Tabela 2 – Area Under the Curve (Área Sob a Curva) (AUC) para os canais isoladamente	50
Tabela 3 – Resultados dos índices de vegetação	51
Tabela 4 – Resultados do pré-processamento	51
Tabela 5 – Resultados fusão <i>early</i>	53
Tabela 6 – Resultados fusão <i>late</i>	53
Tabela 7 – Análise das amostras individuais	53
Tabela 8 – Resumo dos resultados dos Índices de Vegetação para a base de dados 1 e 2	55
Tabela 9 – Resultado dos treinamentos para definição da função de otimização e de perda	57
Tabela 10 – Resultado dos treinamentos para definição do número de camadas das etapas de <i>encode</i> e <i>decode</i>	57
Tabela 11 – Resultado das redes com aplicação de ruído gaussiano na entrada da rede. .	58
Tabela 12 – Resultados arquitetura U-Net para 1 e 2 camadas	58
Tabela 13 – Resultados para arquitetura utilizando blocos residuais	58
Tabela 14 – Resumo dos resultados para os métodos de autoencoder	59
Tabela 15 – Número de amostras por fold para cada base de dados	59
Tabela 16 – Média da AUC para os <i>folds</i> utilizados	60
Tabela 17 – Resumo resultado auto-supervisão	62
Tabela 18 – Resumo dos melhores resultados de todas as etapas do projeto	64

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Trabalhos relacionados	20
1.2	Motivação, Lacuna e Objetivos	22
1.3	Hipótese	23
2	CONCEITOS FUNDAMENTAIS	25
2.1	Considerações iniciais	25
2.2	Agricultura de Precisão	25
2.3	Sensoriamento Remoto	26
2.4	Mapeamento de Vegetação	27
2.4.1	Índices de Vegetação	28
2.5	Métodos de pré-processamento	29
2.6	Deteccção e Avaliação	30
2.7	Aprendizado de Características	31
2.7.1	Autoencoder	32
2.7.2	Funções de custo	34
2.7.3	Otimizadores	34
2.7.3.1	Gradient Descent	34
2.7.3.2	Adaptive learning	35
2.7.4	Variações de arquiteturas autoencoder	35
2.7.4.1	U-Net	35
2.7.4.2	Residual Network	36
2.8	Aprendizado com auto-supervisão	36
2.9	Validação cruzada	37
3	MÉTODO	39
3.1	Considerações iniciais	39
3.2	Base de dados	39
3.3	Índices de vegetação	40
3.4	Pré-processamento	43
3.5	Pipeline de detecção	44
3.6	Aprendizado de índices de vegetação a partir dos dados	44
3.6.1	Diversificação de arquiteturas	46

3.6.2	<i>Auto-supervisão: reconstrução de canais</i>	46
3.7	<i>Avaliação de resultados</i>	47
4	RESULTADOS E DISCUSSÕES	49
4.1	<i>Considerações iniciais</i>	49
4.2	<i>Baseline</i>	49
4.3	<i>Índices de vegetação</i>	50
4.4	<i>Pré-processamento</i>	50
4.5	<i>Fusão</i>	52
4.6	<i>Amostras individuais</i>	53
4.7	<i>Resumo índices de vegetação</i>	54
4.8	<i>Autoencoder</i>	55
4.8.1	<i>Fully connected</i>	55
4.8.2	<i>Convolutacional</i>	55
4.8.2.1	<i>Otimizador e função de custo</i>	56
4.8.2.2	<i>Número de camadas encode/decode</i>	57
4.8.2.3	<i>Denoising autoencoder</i>	57
4.8.2.4	<i>U-Net</i>	58
4.8.2.5	<i>Blocos residuais</i>	58
4.8.3	<i>Resumo Autoencoder</i>	58
4.9	<i>Auto-supervisão pela reconstrução de canais</i>	59
4.9.1	<i>Base de dados 1</i>	59
4.9.2	<i>Base de dados 2</i>	60
4.9.3	<i>Resumo auto-supervisão</i>	60
4.10	<i>Discussão e considerações finais</i>	63
5	CONCLUSÃO E TRABALHOS FUTUROS	65
5.1	<i>Conclusão</i>	65
5.2	<i>Trabalhos Futuros</i>	66
REFERÊNCIAS		67
GLOSSÁRIO		73

INTRODUÇÃO

Recentemente os produtores rurais têm conseguido trazer novas tecnologias para o campo para além do conhecimento agrícola, as quais tem permitido um melhor gerenciamento de suas produções. No processo de trazer estas novas tecnologias para o campo foi desenvolvido o conceito de [Agricultura de Precisão \(AP\)](#), que visa uma melhor compreensão das produções e suas variantes, objetivando o desenvolvimento de produções mais eficientes, com menos recursos e reduzindo custos.

Uma das tecnologias mais utilizadas no contexto da agricultura de precisão é o sensoriamento remoto. Utilizado para aquisição de informações sobre objetos e locais sem contato físico, possui uma ampla gama de modo de captura, utilizando desde dispositivos terrestres até aeronaves, [Veículo Aéreo Não Tripulado \(VANT\)](#)s e satélites, capturando dados que variam desde imagens [RGB](#) até dados hiperespectrais. Com esta variedade de dados suas aplicações abrangem diversas áreas desde o monitoramento ambiental ([THOMAZELLA et al., 2018](#)), detecção de cultivos ([FARIA et al., 2012](#)). Neste trabalho daremos destaque às aplicações para [AP](#) e mapeamento de vegetação ([XIE; SHA; YU, 2008](#)).

A variedade nos dispositivos e sensores para captura afetam diretamente o custo dos projetos, o que, dentre outros fatores, desencoraja o uso da agricultura de precisão ([AUBERT; SCHROEDER; GRIMAUDO, 2012; ADRIAN; NORWOOD; MASK, 2005](#)). Visando aplicações de baixo custo e em países em desenvolvimento, dispositivos de baixa altitude tornam-se uma alternativa viável ao uso de imagens de satélite ([SWAIN; THOMSON; JAYASURIYA, 2010](#)).

Para o desenvolvimento do projeto serão utilizadas bases de dados de imagens capturadas com um dispositivo de baixo custo e baixa altitude ([VERHOEVEN et al., 2009; PONTI, 2013](#)). Esta base, construída com imagens de duas plantações de feijão, permite estudos de mapeamento de vegetação e identificação de regiões que possuem cobertura vegetal. Podendo ser utilizada para classificação binária, onde são identificadas as classes positivas (presença de vegetação) e a classe negativa (ausência de vegetação ou presença de objetos, ervas daninhas, entre outros). As

classes são desbalanceadas, o que implicará na escolha de algumas abordagens específicas no decorrer do trabalho.

1.1 Trabalhos relacionados

Os trabalhos existentes na área de mapeamento e estimativa de vegetação são, em sua maioria, relacionados com base de dados capturadas com câmeras capazes de obter informações do espectro infravermelho (BARATI *et al.*, 2011), (SWAIN; THOMSON; JAYASURIYA, 2010), (XUE; SU, 2017). Porém, câmeras capazes de capturar este tipo de informação, podem ter um custo mais elevado, distanciando-se assim de uma das características que desejamos para o projeto, entretanto não sendo um empecilho para seu funcionamento. O primeiro artigo faz uso de 20 diferentes índices de vegetação, baseados em infravermelho, para construir uma comparação entre as acurácias das estimativas de cobertura vegetal, para cada índice; porém a base de dados utilizada tem como característica ser esparsa, ou seja, com poucas regiões de concentração de vegetação, além de ser capturada por satélite, que apesar de ser capaz de oferecer imagens de alta resolução e com informação de infravermelho, possuem um alto custo para o agricultor, portanto ficando de fora do grupo de índices que utilizaremos em nosso problema. O segundo artigo se aproxima mais do trabalho ao utilizar um helicóptero controlado por rádio, desta forma o custo da criação da base de dados é reduzido. Este helicóptero adquire imagens de baixa altitude, porém também utiliza uma câmera capaz de obter as informações infravermelhas. Por fim o último artigo apresenta uma revisão de diversos índices de vegetação, apresentando seus pontos fortes e fracos, e argumentando da necessidade de cada aplicação conhecer sua base dados, para melhor entender suas limitações e necessidades para poder escolher a melhor abordagem de utilização dos índices, qual índice utilizar e possivelmente como combiná-los.

Entretanto também existem trabalhos que se utilizam de abordagens que não necessitam da informação de infra-vermelho (TORRES-SÁNCHEZ *et al.*, 2014) (PONTI *et al.*, 2016), contendo base de dados de imagens capturadas por dispositivos de baixo custo, e mais acessíveis, tornando assim uma aplicação mais viável para países em desenvolvimento, alinhando-se com uma das características deste projeto. Em ambos os trabalhos o objetivo é estimar a *Vegetation Fraction (Fração de Vegetação) (VF)* em uma plantação. (TORRES-SÁNCHEZ *et al.*, 2014) faz o uso de 6 índices e 2 combinações de índices para quantificar a porção de vegetação; os índices escolhidos pelos autores foram projetados para acentuar o componente verde das imagens, sendo que um em específico foi projetado para lidar com a variação de iluminação durante o dia. Para fins de validação, são capturadas diversas fotos da mesma região. Para a classificação, o autor utiliza um valor fixo de limiar para dar a classe de um pixel, desta forma, após executar a extração do índice de vegetação de uma imagem, o valor resultante para cada pixel é comparado com este limiar pré-definido e caso seja maior é considerado vegetação, caso contrário não é considerado vegetação. A proporção entre pixels classificados como vegetação e total de pixels fornece a VF da imagem. Para validar esta classificação é calculada a acurácia pela diferença da VF encontrada

com a [Observed Vegetation Fraction \(Fração de Vegetação Observada\) \(OVF\)](#). A [OVF](#) é definida como a melhor combinação de foto da região e limiar, sendo que esta melhor combinação é escolhida manualmente com uma interpretação visual do resultado da combinação. Sobre os resultados das acurácias são calculados a acurácia média, o desvio padrão e o coeficiente de variação. Nossos trabalhos começam a se diferenciar logo na escolha do limiar para classificação; enquanto Torres faz uma escolha empírica, em nossa abordagem fazemos um estudo estatístico dos resultados dos índices com o auxílio da curva [ROC](#), o uso da curva [ROC](#) permite uma maleabilidade na classificação, podendo escolher o limiar de forma a ajustarmos nossas taxas de verdadeiro positivo e falso positivo de acordo com o cenário de aplicação. Por fim, como a base de dados utilizada em nossa pesquisa possui o *ground truth*, o cálculo da acurácia se torna mais preciso, uma vez que comparamos com o resultado do índice com a classe real, ficando independente desta análise visual do melhor índice. No artigo de ([PONTI et al., 2016](#)) são apresentadas 2 bases de dados diferentes, uma capturada com informação infra-vermelha e outra sem. A base sem informação de infra-vermelho, esta disponível e foi a mesma utilizada em alguns dos testes apresentados nesta qualificação. Neste artigo o método de classificação utilizado é o [Support Vector Machine \(Máquina de Vetores de Suporte\) \(SVM\)](#) com *kernel Radial Basis Function (Função de Base Radial) (RBF)*, sendo esta uma abordagem válida pois é capaz de fazer uma classificação binária (vegetação e não-vegetação), com um método de treinamento supervisionado. Deste artigo foi possível concluir que o uso de dispositivos de baixo custo e imagens de baixa altitude são viáveis e capazes de trazer benefícios para os agricultores de regiões com menor poder aquisitivo.

Métodos que utilizam aprendizado de representações, em particular aprendizado profundo ([PONTI et al., 2017](#)), tem se mostrado úteis em sensoriamento remoto, ainda que adaptando seus modelos para lidar com os desafios da aplicação ([PENATTI; NOGUEIRA; SANTOS, 2015](#); [MELO et al., 2019](#)). Nesse problema, métodos semi-supervisionados ([SANTOS et al., 2020](#)) ou não supervisionados ([NAKAO; LEVADA, 2020](#)) são vantajosos por reduzir os custos relacionados a anotação de dados. Em particular nessas bases de dados, temos regiões alvo – comumente um ou mais cultivos – com comportamento melhor conhecido e bem comportando, enquanto que as demais regiões possuem maior variância e podem por exemplo ser consideradas anomalias ou um conjunto de dados a ser rejeitado pelo detector ([COSTA; PONTI, 2013](#)). Imagens aéreas para [AP](#) são adquiridas em altas taxas o que pode tornar proibitivo sua rotulação para cada cultivo ou região a ser mapeada, e esse é um dos motivos do sucesso dos métodos baseados em índices de vegetação.

Considerando o cenário em que é preciso obter informações de bases de dados específicas e relativamente pequenas, e uma limitação inerente ao problema de rotulação, os autoencoders, família de métodos não supervisionados de aprendizado de máquina ([VINCENT et al., 2010](#); [LIANG; SHI; ZHANG, 2018](#)), se mostram uma opção viável. Esse métodos vem sendo utilizados no contexto de sensoriamento remoto para extração de características e redução de dimensionalidade de características ([CHENG et al., 2015](#)) ([ZHAO et al., 2015](#)) ([ZHOU et al.,](#)

2015).

Dentre os trabalhos relacionados, o primeiro *paper* (CHENG *et al.*, 2015) utiliza autoencoders como filtragem de dados, tornando-os mais discriminativos. Uma grande diferença entre a abordagem do artigo em relação ao autoencoder e a proposta do nosso projeto está no fato de (CHENG *et al.*, 2015) utilizar o autoencoder para evitar a análise individual dos pixels, enquanto que nossa proposta visa detectar a vegetação justamente no erro da reconstrução de cada pixel. Zhao *et al.* (2015) utiliza o autoencoder como um extrator de características para posteriormente treinar um classificador com regressão logística, aplicado a imagens de sensoriamento remoto geral. Além da aplicação dentro do contexto de sensoriamento remoto serem diferentes da proposta neste projeto, este artigo também se diferencia da proposta pelo fato de utilizar o autoencoder diretamente como *feature* para posterior classificação, enquanto que, como dito anteriormente, nossa proposta é utilizar o erro da reconstrução do autoencoder como índice de vegetação. Isso porque os autoencoders demonstram uma capacidade de se ajustar muito bem ao treinamento, enquanto geram maior erro ou distorcem características de exemplos não vistos (CAVALLARI; RIBEIRO; PONTI, 2018). Nesse projeto exploramos essa aparente limitação em favor de discriminar regiões da imagem relativas a cobertura vegetal. Por fim, Zhou *et al.* (2015) treinou uma rede de autoencoder para utilizar como extrator de características, com as características sendo o resultado da codificação. Novamente temos uma aplicação de autoencoder em um contexto geral de sensoriamento remoto, neste caso o autoencoder é usado diretamente para codificar uma imagem e usar esta codificação como característica, diferentemente da proposta deste projeto, melhor esclarecida no capítulo 3.

1.2 Motivação, Lacuna e Objetivos

Com o avanço e inovações nas áreas de sensoriamento remoto e imageamento um alto volume de dados esta sendo obtido através do sensoriamento remoto, sendo uma técnica muito importante no contexto da AP (QUATAROLI; VICENTE; ARAUJO, 2014). O sensoriamento remoto permite melhorar a produtividade, diminuir impactos ambientais e ainda aumentar a renda dos agricultores. Porém, existe uma variedade muito grande de técnicas que podem ser implantadas nas produções agrícolas, com diferentes abordagens e custos.

Técnicas para detecção de cobertura vegetal baseadas em índices de vegetação possuem formulações dependentes dos canais disponibilizados pelos sistemas de sensoriamento remoto. A decisão do índice de vegetação a ser utilizado esta diretamente ligada ao conhecimento das condições das plantações e dos sistemas de captura. Sistemas de baixa altitude podem sofrer com variações de de iluminação, vento, movimentação do dispositivo, textura da plantação entre outros. Desta forma cada índice resultará um comportamento diferente para diferentes condições da cultura e da captura. Estas técnicas também falham quando comparadas com a anotação manual feita por especialistas, deixando assim uma brecha a ser fechada.

Desta maneira, objetivamos investigar métodos de índices de vegetação, métodos de extração de características, avaliando seus resultados através de análises de acurácia e comparação de desempenho, objetivando realizar pré e pós-processamento de forma a tentar obter o melhor resultado possível a partir dessas abordagens. Adicionalmente, avaliar a viabilidade de técnicas baseadas em arquiteturas de auto-supervisão e autoencoder para produzir um índice de vegetação aprendido a partir dos dados que sejam capazes de suprir a necessidade de selecionar entre índices de vegetação pré-definidos, com objetivo de atingir acurácia igual ou superior a dos índices.

1.3 Hipótese

Considerando o problema da detecção de cobertura vegetal em imagens de baixa altitude, nossa hipótese é formulada em duas partes.

1. Técnicas de pré-processamento das imagens, e pós-processamento de detecções obtidas podem melhorar os resultados dos índices de vegetação aplicados diretamente na imagem adquirida.
2. Uma arquitetura de autoencoder é capaz de aprender uma representação útil a ser utilizada para detecção de cobertura vegetal, utilizando do mapa de erros após o treinamento com exemplos positivos.

Para essa última, temos como premissa que um autoencoder cometerá mais erros nas regiões não vistas, em particular as regiões de classe negativa, que não possui vegetação alvo. Assim, sua saída na forma de um mapa de erros terá informação útil para detectar cobertura vegetal.

CONCEITOS FUNDAMENTAIS

2.1 Considerações iniciais

Neste capítulo serão apresentados os conceitos e temas de maior relevância à pesquisa. Para isso serão apresentados desde os conceitos mais abstratos como Agricultura de Precisão, Sensoriamento Remoto e Mapeamento de Vegetação, que compõe o contexto em que a pesquisa se aplicará. Em seguida virão os conceitos mais técnicos e práticos da pesquisa, como índices de vegetação, métodos de pré-processamento de imagens e técnicas para detecção e avaliação de classificadores binários. Por fim serão apresentados métodos para aprendizado de características, foco da segunda parte desta pesquisa.

2.2 Agricultura de Precisão

A agricultura de precisão se conceitualiza na utilização de tecnologias para observação e cálculos de informações em campo, que sejam capazes de auxiliar o gerenciamento de produções agrícolas, otimizando seus resultados, economizando recursos e reduzindo custos de produção. A utilização de técnica de agricultura de precisão vem com um custo associado, porém, o uso destas tecnologias torna possível ao agricultor realizar análises sobre sua produção, permitindo novas tomadas de decisões, recuperando o investimento na tecnologias através da economia de insumos, como por exemplo: água, fertilizantes e pesticidas.

Conforme representado na Figura 1¹, as etapas ciclo de produção agrícola podem ser enriquecidas com o uso de alguma tecnologia. O controle preciso dos insumos de uma produção afetam todas as etapas de um ciclo de produção tornam a AP fortemente ligada a uma agricultura sustentável, uma vez que quer trazer o melhor para produção, solo e agricultores (BONGIOVANNI; LOWENBERG-DEBOER, 2004). Agricultura sustentável visa criar uma cadeia de

¹ Adaptado de <<http://www.cema-agri.org/page/precision-farming-producing-more-less>> Acesso em Julho de 2018

Figura 1 – Ciclo de Produção

produção sustentável, ecológica e econômica, enquanto que a AP busca estes mesmos objetivos, porém com o monitoramento da variabilidade da área.

A produção agrícola pode ser afetada por diversos fatores, categorizados por (ZHANG; WANG; WANG, 2002) em seis grupos: a) variabilidade de produção, b) variabilidade do campo (aspectos topográficos), c) variabilidade do solo (e fertilidade), d) variabilidade da plantação, e) variabilidade por fatores anômalos, f) variabilidade no gerenciamento. Para cada grupo uma abordagem diferente pode ser tomada. Nesse trabalho assumimos que melhores resultados nas fases (c), (d) e (e), possibilitarão ainda auxílio no gerenciamento, item (f).

2.3 Sensoriamento Remoto

Sensoriamento remoto consiste no exame ou aquisição de informação sobre um objeto ou local a uma certa distância. Estas informações podem ser obtidas por meio de câmeras e sensores que podem estar localizados no solo ou obtidos a partir de sistemas aéreos sub-orbitais como aviões tripulados, *drones* (ou VANT), além de sistemas orbitais como satélites

artificiais. Os sistemas de imageamento aéreo sub-orbitais são comumente mais baratos e possuem características que afetam diretamente sua aplicação e custo, destacando-se a altitude da plataforma de captura, resolução espacial da imagem e a frequência da aquisição. Apesar dos diversos tipos de dados que podem ser obtidos, seu uso se concentra em experimentos que envolvem interpretação e processamento de imagens, permitindo que as imagens sejam manipuladas da forma que for mais conveniente para sua aplicação. O uso de informações obtidas por dispositivos de sensoriamento remoto possibilita aplicação e análise em diversos campos de estudo como: absorção, reflexão e emissão de radiação, produção de mapas, estudos do clima e da atmosfera além de mapeamento e estudo dos recursos naturais. No campo da geologia, o sensoriamento remoto tem sido usado para analisar e mapear áreas de grande cobertura ou de difícil acesso, auxiliando na identificação de rochas e minerais; alguns processos do uso de sensoriamento remoto na geologia são apresentados em (GOETZ; ROWAN, 1981). Informações obtidas com o uso de sensoriamento remoto também são muito utilizadas em aplicações de agricultura de precisão (MULLA, 2013); as informações capturadas são baseadas em radiações refletida, bem como pela fluorescência das plantas (BUSCHMANN; LANGSDORF; LICHTENTHALER, 2000) e emissões térmicas (GOWARD; CRUICKSHANKS; HOPE, 1985; COHEN *et al.*, 2005)

2.4 Mapeamento de Vegetação

A predição da composição vegetal de uma área define o conceito do mapeamento de vegetação. Suas aplicações vão desde auxílio no planejamento de uma restauração ecológica até a avaliar os impactos de uma mudança ambiental nas distribuições da vegetação e de áreas agrícolas.

Técnicas de sensoriamento remoto são capazes de fornecer dados rápidos e confiáveis sobre regiões (XIE; SHA; YU, 2008), que podem ser usados para monitoramento. Uma vez que mudanças na cobertura vegetal, além de condições atmosféricas e de iluminação, resultam diretamente em mudanças nos valores da resposta espectral deste local, é possível mapear valores obtidos em imagens digitais para valores que geram informação útil sobre plantações. A forma mais direta de obter dados sobre a vegetação com base em imagens é o uso de índices de vegetação, descritas na Subseção 2.4.1, promovendo mapeamento de cobertura vegetal com o auxílio do sensoriamento remoto com base na radiação espectral (JR *et al.*, 2003).

2.4.1 Índices de Vegetação

VI consiste em uma transformação matemática que, baseada em bandas² do espectro³, destacando informações que permitam análises espaciais e temporais, estruturais, fenológicos e biofísicos da vegetação (THENKABAIL; SMITH; PAUW, 2000). Suas formulações operam considerando cada pixel da imagem de forma a atuar diretamente sobre os valores das bandas disponíveis, podem levar em conta informações sobre o ambiente dos dados como condições climáticas, tipo de cobertura vegetal ou tipo de solo.

Plantas refletem a luz em comprimentos de onda específicos de acordo com seu estágio de desenvolvimento e saúde. **VI**s utilizam comumente 3 espectros principais: o espectro visível, o qual geralmente é codificado em vermelho, verde e azul (ou RGB, do inglês *red, green, blue*); comprimentos de onda maiores do que a luz visível na região do infravermelho (IR, do inglês *infrared*); e menores do que a luz visível, em particular o ultravioleta (UV, do inglês *ultraviolet*). Esses dados de diferentes intervalos de comprimento de onda são obtidos com sensores multiespectrais, comumente usado em satélites. Na Figura 2a⁴ podemos observar os diferentes comprimentos de onda para seus respectivos espectros, abrangendo o visível, o infravermelho e o ultravioleta. Com o avanço da tecnologia, dispositivos mais sensíveis a diferentes comprimentos de onda do espectro surgiram, permitindo uma maior discriminação entre os comprimentos de onda, como representado na Figura 2b, chamados de sensores hiperespectrais⁵. Com a maior fragmentação no espectro, em abordagens hiperespectrais, cada banda é responsável por um trecho menor de comprimentos de onda, desta forma, para cobrir todos os comprimentos de onda necessários, um maior número de capturas se faz necessário. Cada captura representará uma pequena porção do espaço de frequências, que juntas cobrirão todo o espaço amostral.

A partir da aquisição de dados com sensores nas diferentes faixas descritas, é possível computar os **VI**s, os quais representam um papel importante no contexto de mapeamento de vegetação (LLOYD, 1990; LUCAS; SCHULER, 2007; DUTTA *et al.*, 2016). A aplicação desses índices também inclui o monitoramento de secas (WAN; WANG; LI, 2004) e análise de tendências climáticas (EKLUNDH; OLSSON, 2003).

Deteção de cobertura vegetal: no contexto desta pesquisa deteção de cobertura vegetal será a classificação de um pixel em vegetação alvo, descrita na seção 3.2, ou não-vegetação (qualquer cobertura vegetal que não seja a vegetação alvo). A classificação se dará com a utilização dos valores gerados pelos **VI**s conforme descrito na seção 2.6.

² Banda se refere a um grupo definido de comprimentos de onda dentro de um espectro

³ Diz respeito ao espectro eletromagnético composto por todas as frequências da radiação eletromagnética, que vão desde ondas de baixa frequência, como ondas de rádio, passam pela luz visível e vão até as de maior frequência como a radiação gama

⁴ A Figuras 2a foi adaptada de <<https://courses.lumenlearning.com/boundless-physics/chapter/the-electromagnetic-spectrum/>> Acesso em Outubro de 2020

⁵ A Figura 2b foi adaptada de <<https://www.frontierag.co.uk/blog/what-is-the-difference-between-multispectral-hyperspectral-imagery>> Acesso em Outubro de 2020

(a) Comprimentos de onda

(b) Comparação multi e Hiper espectro

Figura 2 – Representações de diferentes espectros

Fusão de índices de vegetação: além de utilizados individualmente, os índices podem ser combinados para criação de novas métricas (TORRES-SÁNCHEZ *et al.*, 2014). A fusão pode acontecer de duas formas principais. A primeira é chamada de *early fusion*, acontece um passo antes da classificação ou decisão, em que após a extração de diversos VIs o resultado de sua formulação é combinado para gerar um novo conjunto de informações que poderá ser usado para classificação. A segunda forma de combiná-los, *late fusion*, é feita em nível de decisão, ou seja, cada índice é classificado individualmente e o resultado das classificações é combinado.

2.5 Métodos de pré-processamento

Pré-processamento é o nome dado para operações feitas sobre dados brutos, visando prepará-los para um próximo procedimento. Dentre os objetivos do pré-processamento destacam-se: supressão de informações não desejadas, como distorções, ruído e *outliers*, e o realce ou destaque das características relevantes contidas nos dados originais. A análise de dados que não foram devidamente pré-processados podem levar a uma má interpretação dos dados ou dificuldades na obtenção das principais informações contidas nos dados. No caso de imagens, os métodos utilizam a informação local ou contextual como premissas para o pré-processamento. Por exemplo: pixels em uma vizinhança provavelmente pertencem a um mesmo objeto, desta forma, distorções podem ser removidas considerando as informações de sua vizinhança.

Dentre os algoritmos utilizados para pré-processamento de imagens, pode-se destacar aqueles relacionados à redução de ruído baseada em suavização, os quais podem ser escritos pela convolução de um filtro ou *kernel* com uma imagem. A convolução mede a soma dos produtos

entre duas funções(imagem e filtro), ao longo da região de superposição delas em função de um deslocamento que percorrerá a imagem por um todo. Filtros de suavização agem como um filtro passa-baixas, em que componentes de alta frequência são removidos, reduzindo a variância local. A implicação dessa operação é o borramento e degradação de detalhes, texturas, bordas e fronteiras de objetos. Nos casos em que esses detalhes finos são importantes, algoritmos que visam preservar bordas e textura podem ser empregados.

2.6 Detecção e Avaliação

Para realizar a avaliação dos resultados dos VIs pode-se utilizar a análise através da curva ROC (FAWCETT, 2006). A curva ROC consiste em uma representação gráfica do desempenho de um detector ou classificador binário com um limiar de classificação variante. Desta maneira, para um dado de entrada, a curva ROC é produzida analisando a False Positive Rate (Taxa de Falso Positivo) (FPR) e True Positive Rate (Taxa de Verdadeiro Positivo) (TPR) para diferentes limiares. Essa análise é completa no sentido que estima a performance de uma determinada característica (medida como um índice de vegetação, por exemplo) com base em diferentes limiares, permitindo o estudo de um ponto de corte ótimo.

A taxa de falsos positivos expressa a probabilidade de uma amostra negativa ser erroneamente classificada como positiva, enquanto que a taxa de verdadeiros positivos apresenta a chance de uma amostra positiva ser corretamente classificada como positiva; assim um sistema ótimo seria capaz de produzir resultados com 0% de FPR e 100% de TPR. Os valores de FPR e TPR são produzidos pelas fórmulas 2.1 e 2.2; para o cálculo é considerado TP o número de amostras corretamente classificadas como positivas, TN o número de amostras corretamente classificadas como negativas, FN o número de amostras erroneamente classificadas como negativa e FP o número de amostras erroneamente classificadas como positivas.

$$FPR = \frac{FP}{FP + TN} \quad (2.1)$$

$$TPR = \frac{TP}{TP + FN} \quad (2.2)$$

Para obter uma performance global do sistema codificada em um único valor pode-se analisar a AUC da curva ROC. A AUC fornece a probabilidade de um classificador ranquear uma amostra positiva aleatória acima de uma amostra aleatória negativa. Pode ser que, em alguns casos, seja melhor considerar a AUC em determinadas regiões do gráfico, ao invés de toda a área do gráfico (DELONG; DELONG; CLARKE-PEARSON, 1988).

O Equal Error Rate (EER) é também uma medida relevante que pode ser extraída da curva ROC. Ela representa o ponto em que a False Acceptance Rate (Taxa de Falsa Aceitação) (FAR) e a False Rejection Rate (Taxa de Falsa Rejeição) (FRR) são iguais; quanto menor o

EER maior é a acurácia do sistema. **FAR** representa a taxa em que amostras negativas são aceitas como positivas e **FRR** representa a taxa em que amostras positivas são classificadas como negativas. Considerar o valor de **EER** como uma referência para obtenção do limiar de corte para o classificar implica que o sistema não dará prioridade nenhuma das taxas.

Ao utilizar uma base de dados desbalanceadas, a **AUC** pode fornecer uma visão errada da performance do sistema. Assim, adicionalmente, para avaliar bases desbalanceadas utiliza-se do cálculo da acurácia balanceada (do inglês *Balanced Accuracy*)(**BACC**) apresentada pela Equação 2.3, em que P representa o número de amostras positivas e N o número de amostras negativas. A acurácia balanceada permite que a performance seja avaliada para classe, assim se o sistema performar bem em ambas as classes, a acurácia não será afetada, porém, se o sistema obtiver um desempenho ruim em alguma das classes, sua acurácia será penalizada.

$$BACC = \frac{\frac{TP}{P} + \frac{TN}{N}}{2} \quad (2.3)$$

2.7 Aprendizado de Características

O aprendizado de características (do inglês *feature learning*) é o uso de técnicas para permitir que um sistema seja capaz de descobrir autonomamente quais as características relevantes de um conjunto de dados de forma que permita a classificação a partir de dados brutos. Substitui a engenharia de *features*, permite que a máquina aprenda estas características como resultado da otimização de uma função objetivo relacionada a uma tarefa de aprendizado de máquina (**BENGIO; COURVILLE; VINCENT, 2013**).

Existem dois tipos de aprendizados a serem destacados, o aprendizado supervisionado e o não supervisionado. No aprendizado supervisionado os dados fornecidos já estão rotulados, ou seja, os dados de entrada já possuem sua classe definida; desta forma existe uma relação prévia entre a entrada e a saída do sistema, permitindo que o sistema automaticamente se corrija. Já no aprendizado não supervisionado os dados não possuem rótulo, impedindo que estas correções sejam feitas pelo próprio algoritmo; apesar disto esta abordagem permite que problemas com pouco ou nenhum conhecimento prévio sejam estudados (**MELLO; PONTI, 2018**).

Para que uma base de dados seja rotulada é necessário que todos os seus dados passem por um especialista da área para que ele diga qual dado representa qual classe. Porém, do ponto de vista de tempo e custo, esse processo se torna impeditivo; quando buscamos por tecnologias de baixo custo para mapeamento de vegetação, o processo de rotulação fica de lado, necessitando assim de um processo não supervisionado para o seu melhor desenvolvimento.

2.7.1 Autoencoder

Método não supervisionado cujo objetivo é aprender a representar (codificar) dados, comumente com redução de dimensionalidade (Rumelhart; McClelland, 1987). Um autoencoder pode ser construído por meio de camadas densas como no algoritmo **Multilayer Perceptron (MLP)**, e no qual o número de entradas é igual ao de saídas (HINTON; SALAKHUTDINOV, 2006).

O autoencoder possui 2 etapas: a primeira etapa consiste em um codificador (encoder), que aprende a representar a entrada em um outro espaço, comumente de dimensionalidade reduzida ou com restrições de esparsidade, o que induz a redução da dimensionalidade minimizando a perda de informação (PONTI *et al.*, 2017). O resultado do encoder é comumente chamado de **código**, o qual está em um espaço conhecido por **espaço latente**. O encoder pode ser escrito como uma operação sobre uma entrada $\mathbf{x}$ na forma:

$$f(\mathbf{x}) = \mathbf{z}, \quad (2.4)$$

em que $\mathbf{z}$ é o código resultante.

A segunda etapa consiste em um decodificador (decoder) que, a partir do código, aprende a reconstruir os dados de entrada. O decoder pode ser escrito como:

$$g(\mathbf{z}) = \hat{\mathbf{x}}, \quad (2.5)$$

em que $\hat{\mathbf{x}}$ é a entrada reconstruída/estimada.

Quando modelados como redes neurais para processar imagens, os autoencoders (e seus componentes encoder/decoder) podem conter camadas de diferentes tipos, como as densas (*fully connected*), convolucionais, de redução de dimensionalidade espacial (*pooling*) de normalização, entre outras, conforme abaixo.

- **Camadas densas:** São camadas que permitem que todos as unidades de processamento (chamadas de neurônios) da camada anterior se conectam a todos neurônios da próxima camada. Esse tipo de camada pode ser escrita como uma combinação linear dos dados de entrada por neurônio. Para um vetor de entrada de tamanho m , os parâmetros de uma camada densa com k neurônios podem ser representados por uma matriz de pesos $\mathbf{W}$ de tamanho $m \times k$ e um vetor *bias* de tamanho $1 \times k$ dado por $\mathbf{b}$, de forma que a transformação seja obtida por:

$$f(\mathbf{x}) = \mathbf{W}\mathbf{x} + \mathbf{b} = \mathbf{z}, \quad (2.6)$$

- **Camadas convolucionais:** Consiste na aplicação de um filtro (*kernel*) deslizando sobre a entrada. A região que recebe o filtro, chamada de **campo receptivo**, será transformada em um único valor para a próxima camada. Após o filtro percorrer todo o dado de

entrada, o resultado será um **mapa de ativação** ou **mapa de features**. Cada unidade de processamento (relativo a um neurônio na camada convolucional) gera um mapa de ativação i pela convolução entre a imagem de entrada e uma matriz de pesos $\mathbf{h}_i$:

$$f_i(\mathbf{x}) = \mathbf{h}_i * \mathbf{x}, \quad (2.7)$$

- **Redução de dimensionalidade espacial (*pooling*):** Utilizada para dois propósitos: redução de custo computacional e redução de dimensionalidade. Consiste na combinação de conjuntos de neurônios de uma camada em um único neurônio na camada seguinte. Há dois tipos de *pooling* utilizados, *average pooling* e *max pooling*, o primeiro realizando o cálculo da média no conjunto de neurônios de uma camada para a seguinte, enquanto que o segundo extrai o maior valor dentro deste conjunto para a próxima camada.
- **Normalização:** Utilizada para diminuir a covariância entre as unidades da camada, o que acaba afetando o tempo de treinamento.

O aprendizado é feito comumente por meio da função de custo de mínimos quadrados de forma que, se uma imagem $\mathbf{x}$ é fornecida como entrada, e o autoencoder produz a reconstrução $\hat{\mathbf{x}}$, então o custo é:

$$\ell(\mathbf{x}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2. \quad (2.8)$$

Essa função pode ser vista como um erro de reconstrução e pode ser computada por pixel, por imagem, ou por conjunto de imagens, descrito melhor na Subseção 2.7.2.

Como efeito colateral de aprender a reconstrução dos dados de entrada, o autoencoder aprende um espaço latente útil para os dados de treinamento.

Figura 3 – Representação autoencoder para imagens

Na Figura 3⁶ temos uma representação do autoencoder para imagens. Pode-se observar a redução de dimensionalidade que ocorre na transformação da camada de entrada x para a camada oculta/espço latente z . Com a redução de dimensionalidade dados são perdidos, desta forma, por exemplo, o autoencoder pode ser otimizado para remover os ruídos da imagem, aprendendo a reproduzir na camada de saída $\hat{x}$ uma imagem menos ruidosa.

2.7.2 Funções de custo

Usada para avaliar o quão bem uma arquitetura é capaz de modelar um certo dado. Caso uma modelagem não atinja os resultados desejados, a função de custo resultará em um valor alto, que combinado com um otimizador poderá ajudar o algoritmo a melhorar suas previsões. Cada arquitetura e conjunto de dados necessitará de uma função de custo diferente. Podem ser divididos em dois grupos, descritos a baixo:

- **Função custo de regressão:** Utilizado para prever valores contínuos, destacamos a função de **Mean Squared Error (Erro Quadrático Médio) (MSE)**, que possui resultado sempre positivo, sendo uma predição perfeita teria valor 0. O cálculo quadrático pune mais arquiteturas que realizam erros maiores em suas previsões.
- **Função custo de classificação:** Utilizado para prever valores de um conjunto categórico, ou seja, dados que identificam uma categoria. Destacamos a função *Binary Cross-Entropy*, que irá calcular a probabilidade de uma entrada estar em um grupo, sendo o resultado final (classe predita) a média das probabilidades.

2.7.3 Otimizadores

Otimizadores são funções responsáveis por atualizar os pesos nas camadas de uma rede neural e taxas de aprendizado, tentando minimizar a função de custo e definindo como a rede aprenderá.

2.7.3.1 Gradient Descent

O mais básicos dos otimizadores, trabalha de forma iterativa tentando encontrar, em uma curva, a direção de declive máximo, a direção negativa do gradiente, representado na Equação 2.9 por ΔJ . A Figura 4⁷ possui uma representação gráfica do passos de otimização da função, cada marcação na curva representa uma iteração no processo de descida em direção ao ótimo local. Como os pesos são atualizados apenas uma vez após uma iteração completa de todos os dados, os passos dados em direção ao ponto ótimo podem ser grandes demais, pulando os pontos ótimos.

⁶ A Figura 3 foi adaptada de <https://miro.medium.com/max/700/1*5sYuCiuEkCINyyTasfMSA.png> Acesso em Março de 2020

⁷ A Figura 4 foi adaptada de <https://ml-cheatsheet.readthedocs.io/en/latest/gradient_descent.html> Acesso em Outubro de 2020

Na Equação 2.9 apresentamos a fórmula do gradiente descendente onde o próximo passo θ_{n+1} é construído a partir valor atual θ_n menos o produto do gradiente com a taxa de aprendizado α .

$$\theta_{n+1} = \theta_n - \alpha \cdot \Delta J(\theta_n) \quad (2.9)$$

Figura 4 – Representação otimização com Gradiente Descendente

Stochastic Gradient Descent (SGD) é uma variante do gradiente descendente que visa atualizar os parâmetros de forma mais rápida, atualizando a cada amostra ou conjunto de amostras (*batch*) que passam pela rede, diferentemente do gradiente descendente que atualiza apenas ao final de todas as amostras.

2.7.3.2 Adaptive learning

Conjunto de algoritmos como Adagrad, Adadelta, Adam e RMSProp permitem que, diferentemente dos otimizadores baseados no gradiente descendente, cada parâmetro possua uma taxa de aprendizado diferente e adaptativa, influenciando a movimentação do ponto ótimo e permitindo fuga de mínimos locais.

2.7.4 Variações de arquiteturas autoencoder

Além das arquiteturas padrões, baseadas apenas em sequências de camadas de convoluções ou totalmente conectadas, algumas derivações são apresentadas a seguir.

2.7.4.1 U-Net

Derivada da "fully convolutional network" (LONG; SHELHAMER; DARRELL, 2014), projetada para ser capaz de utilizar menos imagens e gerar segmentações mais precisas. O conceito apresentado pelos autores é de que combinando informações das camadas de *pooling* com as camadas de *upsampling* o aprendizado será mais preciso. Na Figura 5 é apresentado um exemplo de arquitetura com as condições da U-Net. Pode-se observar a conexão entre

a camada seguinte ao *max pooling* com a camada anterior ao *upsampling* com o intuito de levar informações espaciais que são reduzidas no decorrer da rede, sendo combinadas com as informações de *features* que são geradas ao final da rede.

Figura 5 – Arquitetura U-Net

2.7.4.2 Residual Network

Baseado na ideia de reaproveitar pesos de uma camada anterior para uma camada a frente (HE *et al.*, 2015). Sendo o bloco residual a base da arquitetura, é composto por 2 ou 3 componentes contendo uma operação de *ReLU* e *batch normalization* entre cada camada, como exemplificado pela figura 6.

Figura 6 – Bloco residual com 2 componentes

2.8 Aprendizado com auto-supervisão

O desenvolvimento de redes com aprendizado baseado em auto-supervisão visa explorar a capacidade de extrair informações inerente aos dados, configurando o aprendizado de forma que a supervisão seja feita pelo próprio dado, tirando a necessidade dos dados serem rotulados (JING; TIAN, 2019). É um subconjunto de métodos de aprendizado não supervisionados, com objetivos como:

- Reconstrução
 - Colorização de imagens
 - Super-resolução
 - Predição de canais
- Tarefas gerais
 - Predição de contexto
 - Transformações geométricas
- Geração automática de rótulos

Na Figura 7 é apresentado uma arquitetura para reconstrução de canais, em que a entrada da rede é uma amostra com um subconjunto de 2 dentre os 3 canais **RGB** da amostra original, e a saída são todos os 3 canais reconstruídos, desta forma o erro é calculado através da comparação da reconstrução dos 3 canais com a imagem original e não com a entrada da rede conforme o autoencoder clássico.

Figura 7 – Arquitetura rede para reconstrução de canais com entrada contendo apenas 2 canais **RGB**

2.9 Validação cruzada

Conjunto de técnicas utilizadas para avaliar a generalização de um conjunto independente de dados em um modelo. Utilizado para avaliar a capacidade de um modelo de avaliar um novo conjunto de dados desconhecidos.

Para execução da validação, os dados de entrada são divididos em conjuntos mutuamente exclusivos, ou seja, não existe um mesmo dados em conjuntos diferentes. O método *holdout* baseia-se na divisão em apenas 2 conjuntos, um para treinamento e um para validação; já o método *k-fold* realiza a divisão dos dados em k conjuntos, dos quais apenas 1 é utilizado para teste e cálculo de acurácia do modelo e os outros $k - 1$ são utilizados para o treinamento. No método *k-fold* o processo de treinamento pode ser realizado k vezes, alterando qual o subconjunto de teste conforme apresentado na Figura 8.

Figura 8 – Validação cruzada - k -fold

MÉTODO

3.1 Considerações iniciais

Neste capítulo é descrita a metodologia desta pesquisa, assim como as duas abordagens, sendo a primeira a melhoria de resultados com o uso de [VI](#) no espectro visível, e a segunda um método de aprendizado auto-supervisionado como forma de estimar um índice a partir dos dados. São apresentadas as bases de dados e suas especificações, os índices de vegetação que serão analisados e comparados, os métodos de pré-processamento estudados, os *pipelines* e suas variações para detecção de cobertura vegetal e, por fim, o desenvolvimento de um autoencoder auto-supervisionado para extração do [VI](#).

3.2 Base de dados

Para o projeto foram utilizadas duas base de dados, ambas adquiridas utilizando o equipamento Skyhook Helikite™ no qual foi acoplada uma câmera digital para obter imagens aéreas de duas plantações de feijão. Este equipamento é um dirigível híbrido de balão de hélio e pipa, resultando em um equipamento de baixo custo para sensoriamento remoto. Para a primeira base de dados a captura foi feita com uma câmera calibrada em relação ao solo, realizando aquisições com um ângulo reto com o solo, enquanto que na segunda base de dados este ângulo não é controlado, permitindo capturas com ângulos variados o que ocasiona mapeamento de regiões de cobertura vegetal fora da área de plantação. Além disso, na segunda base de dados há variação de iluminação, causando descalibramento das bandas.

A câmera presa ao equipamento é uma Fujifilm Z20fd com um [Charge-Coupled Device \(Dispositivo de carga acoplada\) \(CCD\)](#) de 1/2.3 polegadas capaz de capturar imagens de 10 [Megapixel \(MP\)](#). A câmera foi configurada para fazer as capturas com um tempo de exposição de 1/120 segundo e 4.2 f-stop. O dispositivo captura imagens com sensores de espectro visível

(24-bit RGB).

As imagens de ambos conjunto de dados foram obtidas com o dirigível a uma altura de 50 metros em ambas as plantações, 63 dias após emergência das primeiras plantas. As bases possuem imagens de duas cultura com condições de solo diferente. A primeira cultura tendo uma maior produtividade devido a características do solo e a segunda cultura, feita sobre solo compactado, tendo uma produção mais escassa e deixando o solo mais aparente. Cada imagem possui 512×512 pixels, com uma resolução de 1.3cm/pixel.

A primeira base de dados possui 40 imagens, composta por 18 imagens da primeira plantação de feijão, a qual foi cultivada com solo menos compactado, e 22 imagens da segunda plantação, com solo mais compactado conforme disponibilizada por [Ponti et al. \(2016\)](#). A segunda base de dados possui 21 imagens, sendo que não há conhecimento sobre qual imagem é referente a qual plantação. Estas imagens foram capturas em com um sistema descalibrado, ou seja, as imagens possuirão distorções mais severas, rotações, borramentos e mudanças de iluminação. Essa segunda base de dados ainda não foi explorada em trabalhos anteriores. Algumas amostras de cada base são apresentadas nas Figuras 9 e 10.

Todas as imagens das bases possuem uma imagem associada com o seu respectivo *ground truth*, com as classes definidas pela votação da maioria entre 3 especialistas do domínio. As classes são definidas em regiões de 16×16 pixels. Desta forma cada especialista analisou os dados de cada região, decidindo a qual classe esta região pertence. A classe positiva representa pela cor branca, se referencia à cultura de feijão, enquanto que a classe negativa são regiões de solo, vegetação escassa e outro objetos, representados pela cor preta. Devido ao fato da classificação se dar em regiões e não pixel a pixel, os resultados podem ser limitados. As bases de dados são desbalanceadas, com uma proporção de 72.1% de exemplos positivos relativas à cobertura vegetal (num total de 29.556 exemplos) e 27.9% de exemplos negativos referentes à ausência de plantação, solo, outras vegetações e objetos (num total de 11.404 exemplos).

3.3 Índices de vegetação

Como apresentado na Seção 2.4.1 existem diversos tipos de VI, cada qual utiliza informações de diferentes bandas do espectro eletro-magnético. Neste caso os *datasets* utilizados possuem apenas informações do espectro visível, i.e. RGB, o que limite aos VI que consideram apenas as bandas RGB em suas formulações, deixando de fora todas as outras equações dependentes de informações de infravermelho, ultravioleta ou dados hiperespectrais.

Nesse trabalho são investigados 6 VIs diferentes: [Normalized Green-Red Difference Index \(NGRDI\)](#), [Excess Green \(ExG\)](#), [Color Index of Vegetation \(CIVE\)](#), [Vegetativen \(VEG\)](#), [Excess Green minus excess red \(ExGr\)](#), [Woebbecke Index \(WI\)](#), conforme descritos por ([TORRES-SÁNCHEZ et al., 2014](#)) para mapear a presença de vegetação relativa à plantação. As equações e respectivas referências são apresentadas na Tabela 1.

Figura 9 – Amostras base de dados calibrada

Tabela 1 – Índices de vegetação utilizados nesse estudo

Índice	Referência	Equação
NGRDI	(GITELSON <i>et al.</i> , 2002)	$\frac{G-R}{G+R}$
ExG	(WOEBBECKE <i>et al.</i> , 1995)	$2g - r - b$
CIVE	(KATAOKA <i>et al.</i> , 2003)	$0.441r - 0.881g + 0.385b + 18.78745$
VEG	(HAGUE; TILLET; WHEELER, 2006)	$\frac{g}{r^{0.667}b^{0.333}}$
ExGR	(NETO, 2004)	$ExG - 1.4r - g$
WI	(WOEBBECKE <i>et al.</i> , 1995)	$\frac{g-b}{r-g}$

Figura 10 – Amostras base de dados não calibrada

R , G e B são os valores brutos dos componentes vermelho, verde e azul de cada pixel da imagem. Algumas das formulações necessitam que os valores dos pixels estejam normalizados. Quando normalizadas as bandas são dadas pelas mesmas variáveis em minúsculo, conforme abaixo:

$$r = \frac{R}{(R + G + B)}; g = \frac{G}{(R + G + B)}; b = \frac{B}{(R + G + B)} \quad (3.1)$$

Para realizar a análise de *early fusion* sobre os índices são propostas duas combinações utilizando média aritmética e geométrica, apresentadas nas Equações 3.2 e 3.3 respectivamente. Esta combinação de dados é feita antes da classificação dos pixels entre as classes, ou seja, serão extraídos todos os índices, os resultados destes índices são combinados conforme as equações 3.2 e 3.3 e a combinação é classificada. Nas equações, N representa o número de índices de vegetação utilizados e V_i a imagem gerada pelo índice i .

$$ArithmeticMean = \frac{\sum_{i=0}^N V_i}{N} \quad (3.2)$$

$$GeometricMean = \frac{\prod_{i=0}^N V_i}{\prod_{i=0}^N V_i + \prod_{i=0}^N (1 - V_i)} \quad (3.3)$$

Algoritmo 1 – Algoritmo para votação da maioria - *Late Fusion***Assegure:**

A classificação resultante pela votação da maioria

```

1: input: image // Imagem utilizada para extrair os índices de vegetação
2: input: VIs[6] // Vetor com os resultados dos índices de vegetação
3: output: result // Resultado da votação da maioria, mesmo tamanho da imagem original
4: para (x,y) in image.size faça
5:   count = 0
6:   para V in VIs faça
7:     se V(x,y).class == positive então
8:       count ++
9:     fim se
10:  fim para
11:  se count ≥  $\frac{\text{length}(\text{VI}s)}{2}$  então
12:    result(x,y).class = positive
13:  senão
14:    result(x,y).class = negative
15:  fim se
16: fim para

```

Para a abordagem *late fusion*, é feita a combinação do resultado da classificação para cada imagem por meio de votação da maioria, conforme apresentado no Algoritmo 1. O que diferentemente da abordagem anterior, todos os índices são classificados e posteriormente combinados.

3.4 Pré-processamento

Foram investigados 4 métodos de suavização visando suprimir possíveis distorções na imagem, sendo avaliados os seguintes métodos:

- Média (*Blur*): É o método de suavização mais simples dentre os estudados, utiliza um filtro (*kernel*) linear para efetuar a convolução sobre a imagem. Este *kernel* fará com que cada pixel na imagem de entrada seja o resultado da média de seus vizinhos na área do *kernel*, não preservando informações de borda. Este método é muitas vezes utilizados para aproximar o Gaussiano.
- Gaussiano: Similar ao método da média, o método Gaussiano também consiste na convolução de um *kernel* sobre a imagem para efetuar o borramento sem preservar borda, com a única diferença sendo os valores do *kernel*, o qual é construído a partir de uma distribuição gaussiana.
- Mediana: Filtro não-linear capaz de preservar bordas quando o nível de ruído não é muito alto. Para o seu cálculo o filtro ordena todos os valores de pixel na área do *kernel*, cujo elemento central (mediana) será utilizado como novo valor para o pixel.

- Bilateral (TOMASI; MANDUCHI, 1998): Assim como a mediana, é um filtro não-linear capaz de preservar informação de borda. O novo valor do pixel é calculado através da média ponderada dos pixels na área do *kernel*. Semelhante ao método gaussiano com a diferença de que, no método bilateral, os pesos possuem duas componentes; uma usando os mesmos pesos que o filtro Gaussiano, porém o segundo componente considera a diferença de intensidade entre os pixels na vizinhança e o pixel sendo analisado.

3.5 Pipeline de detecção

Definimos um fluxo de processamento de maneira a investigar como as etapas podem afetar a obtenção de mapas de vegetação a partir dos pixels das imagens. Diferentes fluxos são possíveis como apresentando na Figura 11, com duas etapas fundamentais e obrigatórias: a extração de índices e a etapa de detecção/classificação. A etapa de extração consiste em aplicar a formulação do VI, resultando em uma imagem onde cada pixel expressa o resultado deste índice, valores que serão usados para a etapa de detecção/classificação. Alternativamente, o próprio espaço de entrada pode ser utilizado, ou um mapa de características pode ser obtido para esse fim. A seguir, etapa de detecção consiste em aplicar os conceitos apresentados na Seção 2.6 para decisão sobre a classe das regiões. As etapas adicionais, e opcionais, são o de pré-processamento e fusão. A etapa de pré-processamento é responsável por processar a imagem de entrada, enquanto que a etapa de fusão decidirá como será a interação entre os índices de vegetação.

Cada fluxo esta representado na figura por uma seta de cor diferente; o fluxo padrão de reconhecimento é representado pela seta preta, que pode ser alterado com a adição dos passos intermediários, pré-processamento ou fusão, criando novos fluxos de detecção.

Temos como hipótese que a combinação de informações locais via suavização (pré-processamento) pode ajudar a suavizar os resultados dos índices de vegetação. Adicionalmente, de que existe algum tipo de informação complementar entre os diferentes índices de vegetação, e que a combinação dos índices evita o pior caso, i.e. a escolha de um índice inadequado para a tarefa em questão.

3.6 Aprendizado de índices de vegetação a partir dos dados

Com base em métodos de aprendizado de característica não supervisionados, em particular autoencoders e algumas derivações, foram investigadas estratégias para aprender informações de vegetação com base nas imagens disponíveis. Para isso, diferentes arquiteturas e combinações de camadas foram empregadas, e o modelo autoencoder treinado apenas com exemplos da

Figura 11 – Pipeline principal para detecção de cobertura vegetal e passos para melhora nos resultados: setas laranjas indicam pré-processamento da imagens, setas azuis indicam a fusão dos índices de vegetação, e setas verdes indicam a fusão das classificações.

classe positiva, referente à cobertura vegetal. As seguintes arquiteturas de autoencoders foram investigadas:

- unicamente com camadas densas: com 1, 2 e 3 camadas para compor *encoder* e *decoder*;
- unicamente com camadas convolucionais: com 1, 2 e 3 camadas para compor *encoder* e *decoder*.

A escolha dessas arquiteturas visa investigar a contribuição do aprendizado espacial de características promovido pela convolução, em contrapartida ao aprendizado considerando todos os pixels como independentes, no caso das unidades densas.

Além destas arquiteturas foram consideradas técnicas utilizadas nas redes *U-Net* e de bloco residual, apresentados na seção 2.7.4.

Para as arquiteturas básicas de autoencoder foram testados dois cenários: o primeiro cuja a entrada é criada por meio de janelamento de 32×32 pixels, com passo de 8 pixels, sendo o objetivo obter um único valor por janela, e o segundo cuja entrada será obtida fornecendo a imagem completa, com o tamanho original, 512×512 , e caso não seja possível a convergência, redimensionando para 256×256 e 128×128 . O resultado obtido com autoencoders para a imagem completa pode ser utilizado em complemento àquelas informações obtidas com os autoencoders treinados com janelas. O objetivo de estudar essas abordagens é entender se a informação local é suficiente ou se há uma informação global que auxilia no processo de detecção.

O número de neurônios por camada será guiado de acordo com a camada intermediária, a qual deverá ter dimensão N , onde $N \times N$ é a resolução da entrada, e.g. para imagens 32×32 , a camada intermediária terá 32 dimensões. Essa redução está relacionada a raiz quadrada do total de pixels da imagem, função comumente empregada para definir a dimensionalidade da projeção em subespaços.

A hipótese nesse caso é a de que o autoencoder aprenderá as características da vegetação alvo. Os auto encoders usam como função de custo o **MSE** por pixel na imagem, computado conforme abaixo:

$$\mathbf{e} = ||\mathbf{x} - \hat{\mathbf{x}}||^2 \quad (3.4)$$

Intuitivamente, ao avaliar imagens futuras contendo mistura de classes positivas e negativas, fornecendo um mapa no qual regiões com baixo erro médio quadrático são aquelas similares à vegetação utilizada no treinamento, enquanto que as regiões com alto erro são relativas à características espectrais não vistas em tempo de treinamento.

3.6.1 *Diversificação de arquiteturas*

Para analisar o comportamento dos dados para diferentes arquiteturas, são feitas algumas combinações e variações nas arquiteturas de autoencoder padrão, descritos a baixo.

A primeira questão a ser analisada é se a presença do pré-processamento afetará as arquiteturas, assim como os índices de vegetação. Dando início ao estudo das arquiteturas, o primeiro passo é a definição do otimizador e da função de custo a ser utilizadas para os treinamentos, desta forma, sendo fixado 2 camadas de *encode/decode*, as funções serão alternadas e analisadas para encontrar o melhor desempenho. Em seguida o número de camadas para *encode/decode* são variados a fim de encontrar o tamanho de rede que fornecerá a melhor acurácia para os dados. Um estudo sobre a adição de ruído Gaussiano é feito para entender se este tipo de alteração nos dados fornecerá um ganho de desempenho na rede. E, por fim, as arquiteturas serão alteradas para utilizarem de blocos residuais e conceitos da U-Net.

3.6.2 *Auto-supervisão: reconstrução de canais*

Para realizar o treinamento de redes auto-supervisionadas para predição de canais foram realizadas 2 abordagens: a primeira analisando a predição do canal vermelho e a segunda do canal verde, que são os canais mais considerados pelos índices de vegetação apresentados na Seção 3.3.

Conforme representado na Figura 12 o treinamento de ambas as redes são realizados a partir de uma mesma amostra, em que os canais são separados e cada rede receberá um subconjunto de 2 canais, ficando responsável por recriar o canal que não foi introduzido na entrada da rede. Os resultados de cada rede são combinados utilizando o canal reconstruído por cada uma enquanto que o canal azul é a média aritmética da predição das redes.

Figura 12 – Combinação redes para predição de canais

3.7 Avaliação de resultados

Ao longo do desenvolvimento do projeto diferentes abordagens para avaliação dos resultados foi implementada. As principais etapas em que a abordagem de avaliação mudou foram: pipeline para detecção de cobertura vegetal, autoencoders para aprendizado de índices de vegetação e auto-supervisão para reconstrução de canais.

- **Pipeline para detecção de cobertura vegetal:** Para avaliar o desempenho dos índices de vegetação foi construída, diretamente dos resultados dos índices, a curva ROC e extraído os valores de AUC. Para construção da curva o resultados dos índices para todas as amostras são concatenados sequencialmente e este conjunto de dados é utilizado construir o classificador binário.
- **Autoencoder para geração de índice de vegetação:** A avaliação das redes de autoencoder, assim como no método anterior, foi feita atrás da construção da curva ROC e análise da AUC porém os dados para construção são o MSE entre a reconstrução das redes e a amostra de entrada considerando todas as amostras disponíveis.
- **Reconstrução de canal com auto-supervisão:** Assim como os anteriores, a avaliação se baseia na construção da curva ROC, porém ao invés de considerar todas as amostras juntas, eles foram separadas utilizando a técnica de validação cruzada com 5 folds, ou seja, as imagens de entrada foram separadas em 5 grupos e os grupos foram alternados para teste e treinamento seguindo o padrão da validação. Feita através da análise da curva

ROC e da AUC, construída aplicando o melhor índice de vegetação encontrado no erro da reconstrução.

RESULTADOS E DISCUSSÕES

4.1 Considerações iniciais

Neste capítulo são apresentados os resultados obtidos para estudo das base de dados e suas 3 abordagens. A primeira abordagem é dividida em 4 etapas. A primeira etapa apresenta o estudo de cada índice de forma individual, destes resultados será escolhido o índice mais relevante para dar continuidades aos experimentos. O estudo da aplicação de filtros de suavização como método de pré-processamentos é apresentado na segunda etapa. Na terceira etapa a fusão de índices é estudada e seu comportamento é observado. Por fim é apresentado o comportamento dos índices caso cada amostra da base de dados fosse considerada individualmente para a classificação. Na segunda abordagem é afeito um estudo para obtenção de um índice de vegetação com redes de autoencoder. E a última abordagem trata da reconstrução e combinação de canais com redes de auto-supervisão para posterior aplicação de índices de vegetação.

4.2 Baseline

Antes de dar início aos estudos dos índices de vegetação, foi feita uma análise sobre os canais das imagens que estão sendo utilizadas. Desta forma buscamos um valor de referência para a acurácia de cada canal isoladamente. Esta análise é feita da mesma forma que para os índices de vegetação. Sabendo que cada amostra é uma região de 32 x 32 pixels da imagem original, estas amostras são concatenadas para todas as imagens das base de dados e com estes valores extraímos a curva [ROC](#), apresentados na Figura 13 para ambas as base de dados.

Conforme apresentado na Figura 13 e na Tabela 2, temos que para ambas as bases o canal verde aparenta ser o menos significativo, tendo a menor [AUC](#). Para a base de dados 1 o canal vermelho se mostrou o mais significativo, e o canal azul para a base de dados 2.

Figura 13 – RGB ROC plot

Método	Base de dados 1	Base de dados 2
	AUC	
Red	0.816	0.752
Green	0.574	0.641
Blue	0.788	0.814

Tabela 2 – AUC para os canais isoladamente

4.3 Índices de vegetação

O primeiro passo da pesquisa foi entender o comportamento dos índices de vegetação sobre as base de dados utilizadas. Para isso todos os índices foram extraídos e analisados individualmente. Durante a análise observamos a curva ROC de cada índice, observando a relação de desempenho de cada um. Pela figura 14a podemos notar que o índice CIVE se sobressaiu em relação ao demais. Já na figura 14b, referente a base de dados 2, as curvas se entrelaçam muito, não havendo um destaque.

Estes comportamentos ficam evidenciados na Tabela 3, que apresenta a AUC de cada índice junto de sua respectiva acurácia balanceada. Observando os dois principais índices de cada base de dados, temos que o CIVE e o NGRDI obtiveram os melhores resultados para a base 1, enquanto que o WI e o CIVE foram os de melhor performance na base 2. Alguns índices se mostraram eficientes, obtendo AUC acima do melhor canal para cada base de dados, sendo o CIVE acima do melhor caso para ambas as base de dados, desta forma ele será utilizado como base para os próximos testes.

4.4 Pré-processamento

Para a análise de pré-processamento foram aplicados os 4 processos de suavização apresentados na Seção 3.4, separadamente, na duas bases. Para limitar o número de testes o

Figura 14 – VI ROC plot

Tabela 3 – Resultados dos índices de vegetação

Método	Base de dados 1		Base de dados 2	
	AUC	Acurácia	AUC	Acurácia
<i>Canal Vermelho</i>	0.816	-	0.752	-
<i>Canal Azul</i>	0.788	-	0.814	-
NGRDI	0.834	0.739	0.790	0.710
ExG	0.802	0.712	0.801	0.732
CIVE	0.856	0.776	0.838	0.764
VEG	0.798	0.715	0.783	0.711
ExGR	0.826	0.734	0.777	0.698
WI	0.797	0.736	0.835	0.769

Tabela 4 – Resultados do pré-processamento

Filtro	Base de dados 1		Base de dados 2	
	AUC	Acurácia	AUC	Acurácia
Sem suavização	0.856	0.776	0.838	0.764
Média (Blur)	0.882	0.803	0.853	0.780
Gaussiano	0.887	0.795	0.848	0.775
Mediana	0.891	0.807	0.851	0.777
Bilateral	0.870	0.787	0.843	0.770

índice de vegetação escolhido para ser feita a classificação foi o **CIVE**, uma vez que ele foi o método em comum, entre os de melhor desempenho, nas duas bases pela análise anterior. Aplicando o mesmo procedimento de análise, obtemos o gráfico da curva ROC para os métodos de pré-processamento avaliados, apresentados na Figura 15; com seus resultados de AUC e acurácia sumarizado na Tabela 4.

Analisando as bases de dados separadamente, para a primeira base, todos os filtros foram capazes de melhorar a performance do índice, elevando a acurácia de 77.6% para 80.7% no

Figura 15 – Curva ROC para os algoritmos de pré-processamento

melhor caso, que foi utilizando o filtro da mediana. Para a segunda base, os métodos de pré-processamento não foram tão efetivos quanto na primeira, no melhor caso elevando a acurácia de 76.4% para 78.0% com a utilização do filtro da média; os filtros para esta base obtiveram uma performance similar entre si, apesar de todos terem sido capazes de elevar a acurácia.

Analisando a base de dados 1, que é mais cooperativa, o ganho de desempenho com a mediana, filtro que preserva bordas, pode indicar presença de informação relevante para a classificação nestes pontos de transição.

4.5 Fusão

Mantendo o mesmo padrão de análise das etapas anteriores, o processo foi repetido para análise das fusões *early* e *late*, fixando a mediana como filtro de pré-processamento. Os resultados das fusões *early* são apresentados na Tabela 5 e os resultados da fusão *late* na Tabela 6, para ambas as bases.

Na primeira base de dados nenhum dos métodos foi capaz de superar a acurácia do índice CIVE com o filtro da mediana, e no caso da fusão *late* teve uma queda na acurácia de 80.7% para 75.8%. Na fusão *early* os 2 métodos analisados obtiveram um desempenho muito semelhante, com uma diferença de apenas 0.1% no desempenho. Ao contrário do comportamento dos métodos de fusão na base 1, para a base 2 todos os métodos se mostraram eficientes, elevando a acurácia de 77.7% para 80.1% nos métodos de fusão *early*, os quais obtiveram desempenho idêntico.

Pelos resultados da fusão se mostrarem ineficientes para a base de dados 1, porém relevantes para a base de dados 2, podemos concluir que pela dificuldade mais alta de interpretar os dados da base 2, cada índice, separadamente, pode ter sido responsável por extrair uma

Tabela 5 – Resultados fusão *early*

Método	Base de dados 1		Base de dados 2	
	AUC	Acurácia	AUC	Acurácia
Média aritmética	0.881	0.786	0.876	0.801
Média geométrica	0.879	0.785	0.876	0.801

Tabela 6 – Resultados fusão *late*

Método	Base de Dados 1	Base de Dados 2
	Acurácia	Acurácia
Maioria	0.758	0.796

Tabela 7 – Análise das amostras individuais

	Base de Dados 1		Base de Dados 2	
	Média	Desvio Padrão	Média	Desvio Padrão
AUC	0.953	0.029	0.912	0.055
Acurácia	0.891	0.042	0.848	0.062

informação importante para uma dada região, se complementando em si.

4.6 Amostras individuais

Em todos os testes feitos até o momento foi considerada a base inteira para definição do [EER](#) e análise da performance, o que significa que os resultados dos índices de cada imagem foram unificados em um conjunto para extração das informações, desta forma os resultados nos dão uma visão geral do comportamento dos [VIs](#) nas bases de dados. Para realizar a análise das amostras individualmente, as informações de [EER](#) e performance são realizadas imagem a imagem, ao invés de todas juntas. Ao considerar apenas uma imagem por vez, os dados de acurácia obtidos expressam quais seriam as melhores condições para cada imagem, podendo ser considerado como um limite superior e ser usado como referência para os próximos estudos. A média dos valores de acurácia e da [AUC](#) da análise individual das amostras são expressas na Tabela 7.

Ao analisar as imagem individualmente podemos relacionar a acurácia do sistema com a resolução do *ground truth* da base de dados; nestas bases foram consideradas regiões de 16 x 16 pixels para classificação dos pixels, podendo indicar que esta resolução esteja muito alta, que a segmentação das regiões poderia ser feita em áreas menores.

Esta abordagem também permite que analisemos a relação dos métodos de pré-processamento com as imagens. As Figuras 16 e 17 representam o *box plot* da acurácia e da [AUC](#) para cada base de dados. O primeiro *box plot*, em todos os gráficos, representa a análise das imagens sem nenhum pré-processamento e o segundo reflete os resultados com a utilização do filtro da mediana. Utilizando o teste de hipótese de Wilcoxon podemos verificar a relevância estatística

Figura 16 – Boxplot da acurácia e AUC para a base de dados 1: Coluna 1 refere-se às imagens sem pré-processamento, e a 2 às imagens suavizadas

Figura 17 – Boxplot da acurácia e AUC para a base de dados 2: Coluna 1 refere-se às imagens sem pré-processamento, e a 2 às imagens suavizadas

dos dados, pois obtemos $p < 0.05$ em ambos resultados.

4.7 Resumo índices de vegetação

A tabela 8 apresenta o resumo de todos os resultados obtidos durante os estudos dos índices de vegetação. Podemos observar a diferença de comportamento entre as duas base de dados, em que para a primeira base o pré-processamento com um único índice de vegetação teve o melhor desempenho, enquanto que na segunda base o melhor desempenho fica para os métodos de fusão. Destaca-se também que o desempenho com o filtro da mediana indica que a

Método	Base de Dados 1		Base de Dados 2	
	AUC	Acurácia	AUC	Acurácia
Imagens Individuais	0.953±0.029	0.891±0.042	0.912±0.055	0.848±0.062
CIVE	0.856	0.776	0.838	0.764
Mediana	0.891	0.807	0.853	0.780
Média Aritimética	0.881	0.786	0.876	0.801
Média Geométrica	0.879	0.785	0.876	0.801
Maioria	-	0.758	-	0.796

Tabela 8 – Resumo dos resultados dos Índices de Vegetação para a base de dados 1 e 2

borda é importante, e que, apesar disso, o pré-processamento sozinho não é capaz de corrigir problemas de um sistema descalibrado e com distorções severas (base de dados 2), neste caso a fusão passa a ser mais interessante.

4.8 Autoencoder

A seguir serão descritos os resultados das arquiteturas baseadas em autoencoder. Nenhum arquitetura testada foi capaz de convergir quando utilizado a imagem com sua resolução completa, desta maneira todos os resultados apresentados são utilizando o método de janelamento apenas com classes positivas.

4.8.1 Fully connected

A primeira arquitetura a ser avaliada foi a composta apenas por camadas *fully connected*, para 1, 2, e 3 camadas de *encode/decode* alternando o código para 256 e 512.

O treinamento foi realizado utilizando o **MSE** como função de perda por 300 épocas. Todas as 6 redes obtiveram um comportamento semelhante. Durante o treinamento estabilizando nas 100 primeiras épocas conforme exemplificado na Figura 18. Porém, ainda não sendo capaz de reconstruir a imagem corretamente como apresentado pela Figura 19; como o treinamento foi realizado utilizando apenas classes positivas, quando foi introduzida a imagem com classes negativas, a reconstrução falhou, ficando semelhante a da classe positiva (esverdeada).

4.8.2 Convolutacional

Como para os testes com índices de vegetação a aplicação de um pré-processamento levou a um aumento no valor da **AUC**, foram feitos sempre dois treinamentos, o primeiro com o dado original e o segundo com a aplicação do filtro da mediana, que teve o melhor desempenho como descrito na Seção 4.4.

Os primeiros treinamentos foram feitos para decisão do melhor par de funções para otimizador e custo a ser utilizado, para uma rede com uma camada para *encode* e uma camada para *decode*.

Figura 18 – *Loss* para treinamento de autoencoder *fully connected* com 3 camadas e código de tamanho 256

(a) Entrada

(b) Reconstrução

Figura 19 – Resultados rede *Fully Connected*

Definido o par de funções otimizador-custo que construiu a melhor *AUC*, treinamos as redes com 1, 2 e 3 camadas de *encode/decode*.

4.8.2.1 Otimizador e função de custo

Para definição do otimizador e função de custo a serem utilizados, foram analisadas as diferenças de desempenho das *AUCs* para os otimizadores: Adadelta, *SGD* e RMSProp. Para definição da função de custo foram analisadas as funções *binary crossentropy* e *MSE*.

Treinados por 100 épocas, pode-se observar pela Tabela 9 que a presença de uma função de pré-processamento levou a um aumento no valor da *AUC*, assim como nos testes anteriores. O par de função de otimização e de custo que obteve o melhor valor de *AUC* foi *SGD* e *mse*. Desta forma para mantermos estas funções de otimização e perda os treinamento das próximas redes. O melhor otimizador sendo definido pelo *SGD*, o mais simples dentre os 3 testados, pode

indicar que pela base de dados 1 conter poucas variações, os otimizadores mais complexos não são necessários.

Arquitetura		AUC	
Otimizador	Função de custo	Original	Mediana
Adadelta	Binary Crossentropy	0.625	0.648
Adadelta	MSE	0.594	0.607
SGD	Binary Crossentropy	0.762	0.795
SGD	MSE	0.770	0.803
RMSProp	MSE	0.686	0.736

Tabela 9 – Resultado dos treinamentos para definição da função de otimização e de perda

4.8.2.2 Número de camadas encode/decode

Com a definição da função de otimização e de custo, o próximo passo foi definir o número de camadas em cada etapa do *encode* e *decode* mantendo as 100 épocas para treinamento. Foram utilizadas 1, 2 e 3 camadas para cada etapa, com resultados apresentados na Tabela 10. Temos que as redes com 2 e 3 camadas possuem o mesmo desempenho, desta maneira utilizaremos 2 camadas para cada etapa, uma vez que ela possui menos parâmetros e será mais rápida para treinar do que a de 3 camadas. A adição do filtro de pré-processamento manteve o mesmo comportamento que nos testes anteriores, levando a um aumento no valor da AUC.

N-Layers	AUC	AUC - MEDIANA
1 - Layer	0.751	0.785
2 - Layers	0.770	0.803
3 - Layers	0.770	0.803

Tabela 10 – Resultado dos treinamentos para definição do número de camadas das etapas de *encode* e *decode*.

4.8.2.3 Denoising autoencoder

Para analisar se a adição do ruído Gaussiano levaria a um aumento no valor da AUC foi aplicado o ruído com desvio-padrão de 0.1, 0.01, 0.001 e 0.0001 na entrada da rede com otimizador SGD, função de perda MSE e duas camadas de *encode* e *decode*.

Para que houvesse a convergência da rede foi necessário aumentar o número de épocas no treinamento para 300. Apesar do aumento no número de épocas e consequentemente no tempo de treinamento, a adição de ruído Gaussiano aos dados resultou em uma melhora não substancial na AUC. Desta forma não será feita a adição do ruído Gaussiano ao treinamento. Os resultados das redes com ruído Gaussiano estão descritos na Tabela 11.

Noise	AUC	AUC - MEDIANA
0.0001	0.769	0.802
0.001	0.771	0.805
0.01	0.773	0.805
0.1	0.771	0.804

Tabela 11 – Resultado das redes com aplicação de ruído gaussiano na entrada da rede.

4.8.2.4 U-Net

Com a definição do otimizador **SGD**, função de perda **MSE**, 2 camadas de *encode* e *decode* e sem adição de ruído, o melhor valor de **AUC** é 0.770. Para tentar superar este valor foram treinadas duas redes baseadas na arquitetura U-Net para 1 e 2 camadas de codificação. Conforme apresentado na Tabela 12 a nenhuma das arquiteturas foram capazes de superar o valor de **AUC** obtido pela rede anterior.

U-Net	AUC	AUC - MEDIANA
1-Layer	0.756	0.793
2-Layers	0.734	0.772

Tabela 12 – Resultados arquitetura U-Net para 1 e 2 camadas

4.8.2.5 Blocos residuais

Para validar a arquitetura com blocos residuais, foi utilizado a arquitetura padrão com duas camadas, porém as camadas de convolução foram substituídas por blocos residuais. Como apresentado na Tabela 13, a utilização dos blocos residuais levaram a um aumento no valor da **AUC**, de 0.770 na arquitetura padrão de duas camadas para 0.777 com a utilização dos blocos. Os parâmetros de otimizadores, função de custo e épocas foram mantidos.

Arquitetura	AUC	AUC - MEDIANA
Bloco Residual	0.777	0.824
2 - Layers	0.770	0.803

Tabela 13 – Resultados para arquitetura utilizando blocos residuais

4.8.3 Resumo Autoencoder

A tabela 14 contém um resumo melhores resultados obtidos durante os estudos sobre autoencoder. Assim como no estudo dos índices de vegetação, concluímos que o autoencoder também se beneficia de métodos de pré-processamento. Porém o melhor desempenho do autoencoder, se utilizando de blocos residuais com 2 camadas, não apresentou melhoria de desempenho comparado ao melhor resultado dos índices de vegetação. Desta forma não foi realizado esta análise para a base de dados 2.

Método	AUC	
	Original	Mediana
CIVE	0.856	0.891
SGD + MSE - 2 - Layers	0.770	0.803
Denoising (100 épocas)	-	-
Denoising (300 épocas)	0.773	0.805
U-Net - 1-Layer	0.756	0.793
Bloco Residual	0.777	0.824

Tabela 14 – Resumo dos resultados para os métodos de autoencoder

4.9 Auto-supervisão pela reconstrução de canais

Nesta seção são apresentados o desempenho das redes para reconstrução de canal. A avaliação é feita, primeiramente, nos canais separadamente, depois é analisado a combinação das redes através da aplicação do melhor índice de vegetação e construção da ROC. Na Tabela 15 temos o número de amostras para cada fold após o janelamento. Pode-se observar que os *folds* da base de dados 2 possuem um número muito menor de amostras em relação a base de dados 1, o que pode, somado com a não calibração da base de dados 2, dificultar o treinamento.

# Fold	Número de amostras	
	Base de dados 1	Base de dados 2
1	66516	16472
2	68137	15936
3	62872	19055
4	65317	15649
5	63199	14968

Tabela 15 – Número de amostras por fold para cada base de dados

4.9.1 Base de dados 1

Mantendo os parâmetros de treinamento encontrado nos testes anterior produziu os resultados representados na Figura 20. Como para reconstrução do canal verde apenas 1 fold e nenhum *fold* para reconstrução do canal vermelho obtiveram valores acima de 0.770, foi decidido utilizar o otimizador Adam com parâmetros *learning rate* 0.001, 0.9 de decaimento para o primeiro momento e 0.999 para o segundo momento, e MSE como função de custo.

Treinado para 100 épocas com otimizador Adam, na Figura 21a pode-se observar que, pela AUC, a rede para reconstrução do canal verde não obteve um bom desempenho para os *folds* 1 e 2, ficando abaixo de 0.770 enquanto que na Figura 21b os *folds* 3 e 5 quem tiveram o desempenho mais baixo, porém ainda ficando acima de 0.770. Este resultado pode ter relação com o número mais baixo de amostras disponíveis para cada *fold*. Desta forma podemos observar que a mesma arquitetura teve mais facilidade para reconstruir o canal vermelho do que para reconstruir o canal verde. Pela Figura 21c temos que a combinação das redes resultou, com

Figura 20 – Curvas ROC para arquitetura de auto-supervisão com otimizador SGD

Canal	AUC Média	
	Base de dados 1	Base de dados 2
Verde	0.780	0.614
Vermelho	0.857	0.646
Combinação	0.881	0.630

Tabela 16 – Média da AUC para os folds utilizados

exceção do fold 4 em um aumento no valor da AUC. Na Tabela 16 esta representado a média da AUC para os 5 folds testados, e considerando esta média temos que a combinação dos canais leva a um aumento significativo na AUC em relação aos canais individualmente.

4.9.2 Base de dados 2

Assim como para a base de dados 1, o treinamento foi realizado com 300 épocas utilizando o otimizador Adam com os mesmos parâmetros e com função de custo MSE. Diferentemente do primeiro teste, a combinação dos canais não beneficiou os folds. Por exemplo, para o caso do fold 3 temos uma AUC de 0.611 para reconstrução do canal verde e 0.769 para o canal vermelho, porém a combinação dos canais resultou em uma AUC de 0.577, inferior a ambos os valores. Conforme representado na Tabela 16 a combinação dos canais, considerando a média dos folds, levou a uma queda de desempenho quando comparada ao canal vermelho isoladamente.

4.9.3 Resumo auto-supervisão

Conforme exibido na Tabela 17, as base de dados tiveram comportamentos diferentes entre si. A base de dados 1 apresentou ganho quando os canais são combinados, enquanto que a base de dados 2 não obteve. Em ambas as bases a reconstrução do canal vermelho foi a que

Figura 21 – Curvas ROC para arquitetura de auto-supervisão para base de dados 1

obteve melhor desempenho isoladamente. Assim como os estudos de autoencoder, os métodos de auto-supervisão não superam o índice de vegetação.

Na Figura 23 destaca-se a combinação das reconstruções 23b, em que as regiões de não-vegetação ficaram esverdeadas, que esta totalmente relacionada ao CIVE, índice de melhor desempenho pelos testes anteriores. Conforme apresentando na Fórmula 4.1, a componente verde é a de maior peso, ou seja, uma de muita relevância para a formulação, e é negativa, levando a um menor valor de índice para o pixel, o que indica a classe negativa.

$$CIVE = 0.441r - 0.881g + 0.385b + C \quad (4.1)$$

(a) Curva ROC para reconstrução canal verde db2

(b) Curva ROC para reconstrução canal vermelho db2

(c) Canais combinados db2

Figura 22 – Resultados arquitetura auto-supervisão para base de dados 2

Canal	AUC Média	
	Base de dados 1	Base de dados 2
Verde	0.676 ± 0.020	0.614 ± 0.008
Vermelho	0.776 ± 0.011	0.646 ± 0.012
Combinação	0.871 ± 0.002	0.630 ± 0.003
CIVE	0.891	0.851

Tabela 17 – Resumo resultado auto-supervisão

(a) Imagem de entrada

(b) Reconstruções combinadas

(c) Reconstrução canal verde

(d) Reconstrução canal vermelho

Figura 23 – Resultados arquitetura auto-supervisão para base de dados 1

4.10 Discussão e considerações finais

Conforme representado na Tabela 18 o autoencoder para utilização como índice de vegetação, pra seu melhor resultado, não atingiu o melhor resultado utilizando apenas os índices de vegetação. O método de auto-supervisão, para a base de dados 1, quando considerado seu melhor fold foi capaz de superar o índice de vegetação e ficando com uma diferença de 0.010 quando considerado a média dos *folds*. Os resultados da base de dados 2 não foi capaz de atingir, para nenhuma arquitetura, os resultados obtidos pelos índices de vegetação. Desta forma, para a base de dados 2, as arquiteturas testadas não ofereceram nenhum ganho, deixando espaço para estudo de novas arquiteturas ou diminuição no número de folds para que haja mais amostras disponíveis no treinamento.

Canal	AUC	
	Base de dados 1	Base de dados 2
Índice de Vegetação + Suavização	0.891	0.851
Amostras individuais	0.953	0.912
Autoencoder + Blocos Residuais	0.824	-
Melhor <i>Fold</i> - Auto-Supervisão - Combinação	0.947	0.718
Média <i>Folds</i> - Auto-Supervisão - Combinação	0.881	0.630

Tabela 18 – Resumo dos melhores resultados de todas as etapas do projeto

CONCLUSÃO E TRABALHOS FUTUROS

5.1 Conclusão

Nesse trabalho estudamos a utilização de índices de vegetação para mapeamento de vegetação no contexto de sensoriamento remoto com capturas de baixa altitude e baixo custo. Neste estudo foi analisado o comportamento de diferentes **VI**s sobre as bases de dados e estudado como estes índices poderiam ser combinados através de métodos de fusão. Também foi analisado a influência, na classificação, de algoritmos de pré-processamento, especificamente com algoritmos de suavização. Na segunda parte foi estudada uma arquitetura de autoencoder para aprendizado de um índice de vegetação. Por fim foi analisado o uso de arquiteturas auto-supervisionadas para reconstrução de canais e extração de índices de vegetação.

Os resultados apresentados demonstram o comportamento dos índices para cada base, e como eles podem ser combinados com métodos de pré-processamento e fusão para melhorar a detecção de cobertura vegetal. Foi observada significância estatística com resultado positivo para o uso de métodos de pré-processamento, comprovando a primeira hipótese desse trabalho. Adicionalmente, encontramos um resultado de limite superior de desempenho dos índices na base de dados que assume disponibilidade de informação de cada imagem, o que serviu de referência para as próximas etapas. Observamos que as representações obtidas por índices de vegetação são apropriadas para imagens aéreas no contexto da aplicação estudada. Porém, a escolha do melhor índice de forma exaustiva, e o estudo de métodos de pré- e pós-processamento pode ser um processo custoso, o que motivou a investigação de métodos de aprendizado.

O desenvolvimento de uma arquitetura de autoencoder (e variações em suas arquiteturas e unidades de processamento) para aprendizado de um índice de vegetação não se mostrou capaz de atingir o desempenho dos índices de vegetação, principalmente para a base de dados 2, que é mais desafiadora, mostrando resultados muito abaixo dos obtidos por índices de vegetação. O método inspirado em aprendizado auto-supervisionado, que busca aprender uma reconstrução de

um dos canais de cor, por sua vez, mostrou potencial, em particular para a base de dados 1. O mapa de erros resultante do aprendizado dessa reconstrução, quando aplicado em conjunto com um índice de vegetação, destaca ainda mais a diferença entre as regiões de vegetação alvo e as demais, demonstrando aumento na AUC para 3 dos 5 *folds* testados em considerando a média dos *folds*, tendo desempenho muito próximo do melhor caso dos índices de vegetação. Assim, considerando a segunda hipótese, temos que apenas autoencoders que utilizam como tarefa a reconstrução de canais produzem informação para detecção de cobertura vegetal que supera, em alguns cenários, resultados de índices de vegetação.

Considerando todas as abordagens, os índices de vegetação em conjunto com métodos de pré-processamento ainda possuem grande relevância para estimar cobertura vegetal. No entanto, para obter melhorias para além do que é possível com esses índices, há potencial no uso dos erros de reconstrução de canais de cor para uma base de dados calibrada.

5.2 Trabalhos Futuros

As aplicações utilizadas durante o projeto podem ser utilizadas para explorar outras áreas, como diferentes cultivos ou outras aplicações em sensoriamento remoto, como mapeamento de área florestal, desmatamento, áreas urbanas e marítimas. Com relação às bases utilizadas, pode ser estudado o uso de *data augmentation* para aumentar o número de amostras disponíveis. Estudar a combinação dos índices de vegetação com CNN Além disso as arquiteturas já utilizadas podem ser modificadas e adaptadas para explorar outras tarefas de auto-supervisão como uso de pseudo-rótulos, entre outros ou ainda outras abordagens de aprendizado para bases descalibradas, como a base de dados 2 utilizada no projeto.

REFERÊNCIAS

- ADRIAN, A. M.; NORWOOD, S. H.; MASK, P. L. Producers' perceptions and attitudes toward precision agriculture technologies. **Computers and Electronics in Agriculture**, v. 48, n. 3, p. 256 – 271, 2005. ISSN 0168-1699. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0168169905000852>. Citado na página 19.
- AUBERT, B. A.; SCHROEDER, A.; GRIMAUDO, J. It as enabler of sustainable farming: An empirical analysis of farmers' adoption decision of precision agriculture technology. **Decision Support Systems**, v. 54, n. 1, p. 510 – 520, 2012. ISSN 0167-9236. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167923612001972>. Citado na página 19.
- BARATI, S.; RAYEGANI, B.; SAATI, M.; SHARIFI, A.; NASRI, M. Comparison the accuracies of different spectral indices for estimation of vegetation cover fraction in sparse vegetated areas. **Egyptian Journal of Remote Sensing and Space Science**, v. 14, n. 1, p. 49–56, 2011. Cited By 30. Citado na página 20.
- BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, IEEE, v. 35, n. 8, p. 1798–1828, 2013. Citado na página 31.
- BONGIOVANNI, R.; LOWENBERG-DEBOER, J. Precision agriculture and sustainability. **Precision Agriculture**, v. 5, n. 4, p. 359–387, Aug 2004. ISSN 1573-1618. Disponível em: <https://doi.org/10.1023/B:PRAG.0000040806.39604.aa>. Citado na página 25.
- BUSCHMANN, C.; LANGSDORF, G.; LICHTENTHALER, H. Imaging of the blue, green, and red fluorescence emission of plants: An overview. **Photosynthetica**, v. 38, n. 4, p. 483–491, Aug 2000. ISSN 1573-9058. Disponível em: <https://doi.org/10.1023/A:1012440903014>. Citado na página 27.
- CAVALLARI, G.; RIBEIRO, L.; PONTI, M. Unsupervised representation learning using convolutional and stacked auto-encoders: a domain and cross-domain feature space analysis. In: IEEE. **2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.], 2018. p. 440–446. Citado na página 22.
- CHENG, G.; ZHOU, P.; HAN, J.; GUO, L.; HAN, J. Auto-encoder-based shared mid-level visual dictionary learning for scene classification using very high resolution remote sensing images. **IET Computer Vision**, v. 9, n. 5, p. 639–647, 2015. Cited By 15. Citado nas páginas 21 e 22.
- COHEN, Y.; ALCHANATIS, V.; MERON, M.; SARANGA, Y.; TSIPRIS, J. Estimation of leaf water potential by thermal imagery and spatial analysis*. **Journal of Experimental Botany**, v. 56, n. 417, p. 1843–1852, 2005. Disponível em: <http://dx.doi.org/10.1093/jxb/eri174>. Citado na página 27.
- COSTA, G. B.; PONTI, M. Green coverage detection on sub-orbital plantation images using anomaly detection. In: SPRINGER. **Iberoamerican Congress on Pattern Recognition**. [S.l.], 2013. p. 92–99. Citado na página 21.

- DELONG, E. R.; DELONG, D. M.; CLARKE-PEARSON, D. L. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. **Biometrics**, JSTOR, p. 837–845, 1988. Citado na página 30.
- DUTTA, D.; DAS, P. K.; ALAM, K. A.; SAFWAN, P.; PAUL, S.; NANDA, M. K.; DADHWAL, V. K. Delta area at near infrared region ($text{DA}_{text{NIR}}$) x2014; a novel approach for green vegetation fraction estimation using field hyperspectral data. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, v. 9, n. 9, p. 3970–3981, Sept 2016. ISSN 1939-1404. Citado na página 28.
- EKLUNDH, L.; OLSSON, L. Vegetation index trends for the african sahel 1982–1999. **Geophysical Research Letters**, Wiley Online Library, v. 30, n. 8, 2003. Citado na página 28.
- FARIA, F. A.; SANTOS, J. A. dos; TORRES, R. d. S.; ROCHA, A.; FALCAO, A. X. Automatic fusion of region-based classifiers for coffee crop recognition. In: IEEE. **2012 IEEE International Geoscience and Remote Sensing Symposium**. [S.l.], 2012. p. 2221–2224. Citado na página 19.
- FAWCETT, T. An introduction to roc analysis. **Pattern recognition letters**, Elsevier, v. 27, n. 8, p. 861–874, 2006. Citado na página 30.
- GITELSON, A.; KAUFMAN, Y.; STARK, R.; RUNDQUIST, D. Novel algorithms for remote estimation of vegetation fraction. **Remote Sensing of Environment**, v. 80, n. 1, p. 76–87, 2002. Cited By 491. Citado na página 41.
- GOETZ, A. F.; ROWAN, L. C. Geologic remote sensing. **Science**, American Association for the Advancement of Science, v. 211, n. 4484, p. 781–791, 1981. Citado na página 27.
- GOWARD, S. N.; CRUICKSHANKS, G. D.; HOPE, A. S. Observed relation between thermal emission and reflected spectral radiance of a complex vegetated landscape. **Remote Sensing of Environment**, v. 18, n. 2, p. 137 – 146, 1985. ISSN 0034-4257. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0034425785900446>>. Citado na página 27.
- HAGUE, T.; TILLET, N.; WHEELER, H. Automated crop and weed monitoring in widely spaced cereals. **Precision Agriculture**, v. 7, n. 1, p. 21–32, 2006. Cited By 84. Citado na página 41.
- HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. **CoRR**, abs/1512.03385, 2015. Disponível em: <<http://arxiv.org/abs/1512.03385>>. Citado na página 36.
- HINTON, G. E.; SALAKHUTDINOV, R. R. Reducing the dimensionality of data with neural networks. **science**, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006. Citado na página 32.
- JING, L.; TIAN, Y. **Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey**. 2019. Citado na página 36.
- JR, P. J. P.; HATFIELD, J. L.; SCHEPERS, J. S.; BARNES, E. M.; MORAN, M. S.; DAUGHTRY, C. S.; UPCHURCH, D. R. Remote sensing for crop management. **Photogrammetric Engineering & Remote Sensing**, American Society for Photogrammetry and Remote Sensing, v. 69, n. 6, p. 647–664, 2003. Citado na página 27.
- KATAOKA, T.; KANEKO, T.; OKAMOTO, H.; HATA, S. Crop growth estimation system using machine vision. In: . [S.l.: s.n.], 2003. v. 2, p. 1079–1083. Cited By 86. Citado na página 41.

LIANG, P.; SHI, W.; ZHANG, X. Remote sensing image classification based on stacked denoising autoencoder. **Remote Sensing**, Multidisciplinary Digital Publishing Institute, v. 10, n. 1, p. 16, 2018. Citado na página 21.

LLOYD, D. A phenological classification of terrestrial vegetation cover using shortwave vegetation index imagery. **International Journal of Remote Sensing**, Taylor Francis, v. 11, n. 12, p. 2269–2279, 1990. Disponível em: <<https://doi.org/10.1080/01431169008955174>>. Citado na página 28.

LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. **CoRR**, abs/1411.4038, 2014. Disponível em: <<http://arxiv.org/abs/1411.4038>>. Citado na página 35.

LUCAS, A. d. A.; SCHULER, C. A. Análise do ndvi/noaa em cana-de-açúcar e mata atlântica no litoral norte de pernambuco, brasil analysis of ndvi/noaa in sugar cane and atlantic forest in the north of pernambuco state, brazil. **Revista Brasileira de Engenharia Agrícola e Ambiental-Agriambi**, Directory of Open Access Journals, v. 11, n. 6, p. 607–614, 2007. Citado na página 28.

MELLO, R. F.; PONTI, M. A. **Machine Learning: A Practical Approach on the Statistical Learning Theory**. [S.l.]: Springer, 2018. Citado na página 31.

MELO, V. V. de; SOTTO, L. F.; LEONARDO, M. M.; FARIA, F. A. Automatic meta-feature engineering for cnn fusion in aerial scene classification task. **IEEE Geoscience and Remote Sensing Letters**, IEEE, 2019. Citado na página 21.

MULLA, D. J. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. **Biosystems engineering**, Elsevier, v. 114, n. 4, p. 358–371, 2013. Citado na página 27.

NAKAO, E. K.; LEVADA, A. L. Unsupervised learning and feature extraction in hyperspectral imagery. In: SPRINGER. **International Conference on Computational Science and Its Applications**. [S.l.], 2020. p. 792–806. Citado na página 21.

NETO, J. C. p. 1–170, 2004. Cited By 1. Citado na página 41.

PENATTI, O. A.; NOGUEIRA, K.; SANTOS, J. A. D. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In: **Proceedings of the IEEE conference on computer vision and pattern recognition workshops**. [S.l.: s.n.], 2015. p. 44–51. Citado na página 21.

PONTI, M.; CHAVES, A.; JORGE, F.; COSTA, G.; COLTURATO, A.; BRANCO, K. Precision agriculture: Using low-cost systems to acquire low-altitude images. **IEEE Computer Graphics and Applications**, v. 36, n. 4, p. 14–20, 2016. Cited By 0. Citado nas páginas 20, 21 e 40.

PONTI, M. A.; RIBEIRO, L. S. F.; NAZARE, T. S.; BUI, T.; COLLOMOSSE, J. Everything you wanted to know about deep learning for computer vision but were afraid to ask. In: IEEE. **Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2017 30th SIBGRAPI Conference on**. [S.l.], 2017. p. 17–41. Citado nas páginas 21 e 32.

PONTI, M. P. Segmentation of low-cost remote sensing images combining vegetation indices and mean shift. **IEEE Geoscience and Remote Sensing Letters**, IEEE, v. 10, n. 1, p. 67–70, 2013. Citado na página 19.

- QUATAROLI, C. F.; VICENTE, L. E.; ARAUJO, L. S. d. Sensoriamento remoto. **Capítulo em Livro Técnico-Científico**, Embrapa Territorial, p. 61–79, 2014. Citado na página 22.
- Rumelhart, D. E.; McClelland, J. L. Learning internal representations by error propagation. In: _____. **Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations**. [S.l.: s.n.], 1987. p. 318–362. Citado na página 32.
- SANTOS, F. P. D.; ZOR, C.; KITTLER, J.; PONTI, M. A. Learning image features with fewer labels using a semi-supervised deep convolutional network. **Neural Networks**, Elsevier, v. 132, p. 131–143, 2020. Citado na página 21.
- SWAIN, K.; THOMSON, S.; JAYASURIYA, H. Adoption of an unmanned helicopter for low-altitude remote sensing to estimate yield and total biomass of a rice crop. **Transactions of the ASABE**, v. 53, n. 1, p. 21–27, 2010. Cited By 61. Citado nas páginas 19 e 20.
- THENKABAIL, P.; SMITH, R.; PAUW, E. D. Hyperspectral vegetation indices and their relationships with agricultural crop characteristics. **Remote Sensing of Environment**, v. 71, p. 158–182, 02 2000. Citado na página 28.
- THOMAZELLA, R.; CASTANHO, J.; DOTTO, F. R.; JÚNIOR, O. R.; ROSA, G. H.; MARRANA, A. N.; PAPA, J. P. Environmental monitoring using drone images and convolutional neural networks. In: IEEE. **IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium**. [S.l.], 2018. p. 8941–8944. Citado na página 19.
- TOMASI, C.; MANDUCHI, R. Bilateral filtering for gray and color images. In: IEEE. **Computer Vision, 1998. Sixth International Conference on**. [S.l.], 1998. p. 839–846. Citado na página 44.
- TORRES-SÁNCHEZ, J.; PEÑA, J.; CASTRO, A. D.; LÓPEZ-GRANADOS, F. Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from uav. **Computers and Electronics in Agriculture**, Elsevier, v. 103, p. 104–113, 2014. Citado nas páginas 29 e 40.
- TORRES-SÁNCHEZ, J.; PEÑA, J.; CASTRO, A. de; LÓPEZ-GRANADOS, F. Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from uav. **Computers and Electronics in Agriculture**, v. 103, p. 104–113, 2014. Cited By 69. Citado na página 20.
- VERHOEVEN, G. J.; LOENDERS, J.; VERMEULEN, F.; DOCTER, R. Helikite aerial photography—a versatile means of unmanned, radio controlled, low-altitude aerial archaeology. **Archaeological Prospection**, Wiley Online Library, v. 16, n. 2, p. 125–138, 2009. Citado na página 19.
- VINCENT, P.; LAROCHELLE, H.; LAJOIE, I.; BENGIO, Y.; MANZAGOL, P.-A.; BOTTOU, L. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. **Journal of machine learning research**, v. 11, n. 12, 2010. Citado na página 21.
- WAN, Z.; WANG, P.; LI, X. Using modis land surface temperature and normalized difference vegetation index products for monitoring drought in the southern great plains, usa. **International journal of remote sensing**, Taylor & Francis, v. 25, n. 1, p. 61–72, 2004. Citado na página 28.

WOEBBECKE, D.; MEYER, G.; BARGEN, K. V.; MORTENSEN, D. Color indices for weed identification under various soil, residue, and lighting conditions. **Transactions of the American Society of Agricultural Engineers**, v. 38, n. 1, p. 259–269, 1995. Cited By 391. Citado na página 41.

XIE, Y.; SHA, Z.; YU, M. Remote sensing imagery in vegetation mapping: a review. **Journal of Plant Ecology**, v. 1, n. 1, p. 9–23, 2008. Disponível em: <<http://dx.doi.org/10.1093/jpe/rtm005>>. Citado nas páginas 19 e 27.

XUE, J.; SU, B. Significant remote sensing vegetation indices: A review of developments and applications. **Journal of Sensors**, v. 2017, 2017. Cited By 1. Citado na página 20.

ZHANG, N.; WANG, M.; WANG, N. Precision agriculture—a worldwide overview. **Computers and Electronics in Agriculture**, v. 36, n. 2, p. 113 – 132, 2002. ISSN 0168-1699. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0168169902000960>>. Citado na página 26.

ZHAO, W.; GUO, Z.; YUE, J.; ZHANG, X.; LUO, L. On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery. **International Journal of Remote Sensing**, v. 36, n. 13, p. 3368–3379, 2015. Cited By 24. Citado nas páginas 21 e 22.

ZHOU, W.; SHAO, Z.; DIAO, C.; CHENG, Q. High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder. **Remote Sensing Letters**, Taylor Francis, v. 6, n. 10, p. 775–783, 2015. Disponível em: <<http://dx.doi.org/10.1080/2150704X.2015.1074756>>. Citado na página 22.

GLOSSÁRIO

AP Agricultura de Precisão.

AUC Area Under the Curve (Área Sob a Curva).

CCD Charge-Coupled Device (Dispositivo de carga acoplada).

CIVE Color Index of Vegetation.

EER Equal Error Rate.

ExG Excess Green.

ExGr Excess Green minus excess red.

FAR False Acceptance Rate (Taxa de Falsa Aceitação).

FPR False Positive Rate (Taxa de Falso Positivo).

FRR False Rejection Rate (Taxa de Falsa Rejeição).

MLP Multilayer Perceptron.

MP Megapixel.

MSE Mean Squared Error (Erro Quadrático Médio).

NGRDI Normalized Green-Red Difference Index.

OVF Observed Vegetation Fraction (Fração de Vegetação Observada).

RBF Radial Basis Function (Função de Base Radial).

RGB Red, Green, Blue (Vermelho, Verde, Azul).

ROC Receiver Operating Characteristic (Característica de Operação do Receptor).

SGD Stochastic Gradient Descent.

SVM Support Vector Machine (Máquina de Vetores de Suporte).

TPR True Positive Rate (Taxa de Verdadeiro Positivo).

VAE Variational Autoencoder (Autoencoder variacionais).

VANT Veículo Aéreo Não Tripulado.

VEG Vegetativen.

VF Vegetation Fraction (Fração de Vegetação).

VI Vegetation Index (Índice de Vegetação).

WI Woebbecke Index.