



CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

INTRODUÇÃO AO PROCESSAMENTO DIGITAL DE IMAGENS

Detector de Nuvens em Imagens de Satélite

Professor:

Leonardo Vidal Batista

*Departamento de Sistemas
de Computação*

Autores:

João Victor Silva Bezerra Nascimento
20200007760

André Vitor Oliveira Brito
20210066943

9 de maio de 2024

1 Introdução

A observação e análise de padrões atmosféricos desempenham um papel fundamental na compreensão e previsão do clima global. Entre esses padrões, as nuvens desempenham um papel crucial, influenciando os padrões de temperatura, precipitação e radiação solar na atmosfera terrestre. Portanto, a detecção precisa e eficiente de nuvens é essencial para uma variedade de aplicações, incluindo meteorologia, monitoramento ambiental, estudos climáticos e sensoriamento remoto.

Neste contexto, o avanço da tecnologia de sensoriamento remoto e a disponibilidade de grandes conjuntos de dados climáticos têm impulsionado o desenvolvimento de técnicas de detecção de nuvens baseadas em inteligência artificial. Estas técnicas visam identificar e classificar diferentes tipos de nuvens a partir de imagens de satélite e outras fontes de dados, automatizando o processo de análise e permitindo uma compreensão mais detalhada dos fenômenos atmosféricos.

O presente trabalho tem como objetivo explorar e avaliar métodos de detecção de nuvens utilizando o dataset **38-Cloud**[3][2], um conjunto de dados abrangente que contém imagens de satélite multiespectrais e suas respectivas máscaras de segmentação de nuvens. A análise e modelagem desses dados serão realizadas utilizando técnicas de aprendizado de máquina, com o intuito de desenvolver um modelo preciso e robusto, capaz de identificar e classificar nuvens em diferentes cenários atmosféricos.

Ao compreender e aprimorar as técnicas de detecção de nuvens, espera-se contribuir para uma melhor compreensão dos processos climáticos globais, auxiliando na previsão do tempo, monitoramento ambiental e estudos científicos sobre as mudanças climáticas. Além disso, a aplicação prática dessas técnicas pode beneficiar uma ampla gama de setores, incluindo agricultura, energia renovável, aviação e gestão de desastres naturais.

2 Revisão Bibliográfica

2.1 Métodos Tradicionais de Detecção de Nuvens

Historicamente, métodos tradicionais de detecção de nuvens baseados em técnicas de segmentação de imagens e processamento digital têm sido amplamente utilizados. Estes métodos geralmente envolvem a extração de características visuais das imagens, como textura, cor e forma, seguida da aplicação de algoritmos de segmentação, como limiarização, regiões de crescimento e transformada de Hough. Além disso, vale ressaltar a **FMask**, que é uma das técnicas que mais se destaca na identificação de nuvens usando apenas processamento puro de imagens.

2.2 Aprendizado de Máquina na Detecção de Nuvens

Com o avanço da tecnologia de sensoriamento remoto e o aumento da disponibilidade de grandes conjuntos de dados climáticos, técnicas baseadas em aprendizado de máquina têm ganhado destaque na detecção de nuvens. Algoritmos de classificação supervisionada, como máquinas de vetores de suporte (SVM), redes neurais convolucionais (CNN) e árvores de decisão, têm sido amplamente utilizados para identificar padrões complexos nas imagens de satélite e distinguir entre diferentes tipos de nuvens.

Além de métodos de aprendizado de máquina tradicionais, técnicas mais recentes e especializadas, como o **Cloud-Net**[3][2] e o **YOLOv8**, têm se destacado na área de segmentação de nuvens. O Cloud-Net é conhecido por sua precisão e eficiência na identificação e classificação de nuvens em imagens de satélite. Por outro lado, o YOLOv8 é reconhecido por sua capacidade de realizar detecção de objetos em tempo real, incluindo nuvens, com alta velocidade e desempenho.

Essas abordagens avançadas permitem uma análise mais detalhada das características das nuvens, incluindo sua forma, textura e padrões de movimento, contribuindo para uma melhor compreensão e monitoramento dos fenômenos atmosféricos.

2.3 Desafios e Tendências Futuras

Apesar dos avanços recentes, a detecção de nuvens continua a enfrentar vários desafios, incluindo a presença de artefatos de sensoriamento, oclusões parciais e a complexidade das condições atmosféricas. Para superar esses desafios, pesquisas futuras estão explorando o uso de abordagens multimodais, integração de dados de diferentes fontes e o desenvolvimento de algoritmos mais robustos e escaláveis.

3 Metodologia

3.1 Ferramentas Utilizadas

PyCaret[4] (Classification) Uma biblioteca de aprendizado de máquina automatizada para simplificar o processo de construção, treinamento e avaliação de modelos de classificação.

Pandas Uma poderosa biblioteca de análise de dados que oferece estruturas de dados flexíveis e ferramentas para manipulação e análise de dados.

NumPy Uma biblioteca fundamental para computação científica que fornece suporte para arrays e matrizes multidimensionais, juntamente com funções matemáticas de alto nível para operar nesses arrays.

Tiffle Uma biblioteca para leitura e escrita de arquivos de imagem no formato TIFF, com suporte para várias plataformas e tipos de dados.

OpenCV Uma biblioteca de visão computacional com uma ampla gama de funções para processamento de imagens e vídeos, incluindo operações básicas, detecção de objetos, reconhecimento de padrões e muito mais.

SciPy Uma biblioteca que fornece funções e algoritmos matemáticos de alto nível para computação científica, incluindo otimização, álgebra linear, integração, interpolação, transformadas de Fourier e muito mais.

Google Colab É uma plataforma de computação em nuvem gratuita fornecida pelo Google. Ele oferece um ambiente de Jupyter Notebook interativo que permite escrever e executar código Python diretamente no navegador da web.

3.2 Dataset

O uso de conjuntos de dados de alta qualidade desempenha um papel fundamental no desenvolvimento e avaliação de algoritmos de detecção de nuvens. Nesta seção, é apresentada uma visão geral do conjunto de dados utilizado neste trabalho, destacando suas características principais e sua relevância para a pesquisa em detecção de nuvens.

3.2.1 Origem e Características

O conjunto de dados utilizado foi o *”38-Cloud: Cloud Segmentation in Satellite Images”*, que contém 38 imagens tiradas pelo satélite **Landsat 8**, e suas máscaras de segmentação de nuvens extraídas manualmente à nível de *pixel*. O conjunto de dados utilizado foi coletado a partir da **Kaggle**, uma plataforma online que oferece uma ampla variedade de conjuntos de dados públicos em diversas áreas.

Cada imagem do conjunto de dados é acompanhada por uma máscara de segmentação de nuvens, que identifica as regiões de nuvens na imagem utilizando anotações manuais.

Nessas máscaras de nuvem, cada pixel da imagem é rotulado como “nuvem” ou “não nuvem”, facilitando a análise e avaliação de algoritmos de detecção de nuvens.

3.2.2 Divisão em Conjuntos de Treinamento e Teste

O conjunto de dados foi dividido em conjuntos de treinamento e teste para fins de desenvolvimento e avaliação de modelos de detecção de nuvens. A divisão foi realizada de forma a garantir uma distribuição equilibrada de imagens e máscaras de nuvens em ambos os conjuntos, garantindo uma representação adequada das condições atmosféricas e tipos de nuvens presentes nas imagens.

O conjunto de treinamento é utilizado para treinar e ajustar os modelos de detecção de nuvens, enquanto o conjunto de teste é utilizado para avaliar o desempenho dos modelos em dados não vistos. Esta abordagem ajuda a garantir que os modelos sejam capazes de generalizar para novas imagens e condições atmosféricas, tornando-os mais robustos e confiáveis em aplicações do mundo real.

3.2.3 Organização dos dados

As imagens inteiras dessas cenas são recortadas em múltiplos patches de 384*384 para serem adequados para algoritmos de segmentação semântica baseados em aprendizado profundo. Existem 8400 patches para treinamento e 9201 patches para teste. Cada patch possui 4 canais espectrais correspondentes que são Vermelho (banda 4), Verde (banda 3), Azul (banda 2) e Infravermelho Próximo (banda 5). Ao contrário de outras imagens de visão computacional, esses canais não são combinados. Em vez disso, eles estão em seus diretórios correspondentes.

3.2.4 Disponibilidade e Acesso

O conjunto de dados completo, juntamente com as máscaras de segmentação de nuvens, está disponível gratuitamente para pesquisa e desenvolvimento na plataforma *Kaggle*.

3.3 Pré-processamento dos Dados

Para alimentar os modelos, foram escolhidos patches de imagens de treinamento, que tiveram pixels de cada canal carregados separadamente em *DataFrames*, passando então por uma concatenação para formar a estrutura final de alimentação do *PyCaret*. Além disso, foi feito um treinamento balanceado dos modelos, de forma que somente um subconjunto de cada patch foi carregado, escolhendo entre pixels com nuvem ou sem nuvem.

3.4 Treinamento do Modelo

3.4.1 Modelo 1

Para o Modelo 1, foram selecionados aleatoriamente 2 patches de máscaras de segmentação com pixels “com nuvem” e “sem nuvem”, juntamente com seus respectivos canais de banda vermelho, verde, azul e infravermelho. No total o modelo foi treinado com 5 colunas 294912 *pixels* cada.

3.4.2 Modelo 2

Semelhante ao Modelo 1, para o Modelo 2 foram selecionados aleatoriamente 5 patches de máscaras de segmentação com pixels "com nuvem" e "sem nuvem", juntamente com seus respectivos canais de banda. Esta ampliação do número de patches utilizados visou melhorar a capacidade de generalização do modelo. Com isso o modelo foi treinado com 5 colunas 737.280 *pixels*.

3.4.3 Modelo 3

Para o Modelo 3, além da seleção aleatória de patches, foram aplicados critérios adicionais de exclusão de imagens que continham gelo. Isso foi feito para facilitar o processo de identificação das nuvens e melhorar a precisão do modelo. Além disso, foi utilizado um algoritmo de balanceamento para garantir a mesma quantidade de pixels com e sem nuvem na imagem, contribuindo para um conjunto de dados mais equilibrado. Desse modo, o modelo 3, por mais que tenha sido treinado com mais patches, por conta do critério de seleção de pixels, ficou com 341026 *pixels* em cada coluna.

3.4.4 Modelo 4

Similar ao Modelo 3, para o Modelo 4 foram selecionadas imagens livres de gelo e aplicado um algoritmo de balanceamento para garantir uma representação equilibrada de pixels com e sem nuvem. No entanto, neste caso, foram selecionados 14 patches para aumentar a diversidade e a cobertura do conjunto de dados. No total o modelo foi treinado com 5 colunas 938860 *pixels* cada.

4 Resultados

4.1 Método de Avaliação

Para avaliar as previsões dos modelos, foi utilizada a técnica *Intersection over Union (IoU)*. O IoU é uma métrica utilizada para avaliar a precisão de detecções de objetos em tarefas de visão computacional, como detecção de nuvens em imagens de satélite. O IoU mede a sobreposição entre duas áreas retangulares, chamadas de *bounding boxes*.

Para calcular o IoU, primeiro determina-se a área de interseção entre duas bounding boxes, que é a região onde os dois retângulos se sobrepõem. Em seguida, calcula-se a área da união das duas bounding boxes, que é a soma das áreas das duas caixas, menos a área de interseção. Por fim, dividimos a área de interseção pela área da união para obter o valor do IoU.

Ao aplicar o IoU na avaliação de desempenho de modelos de detecção de nuvens, cada imagem contém previsões de várias nuvens, representadas por bounding boxes. Para associar corretamente as previsões com as nuvens reais presentes na imagem (ground-truth), é necessário calcular o IoU individualmente entre cada bounding box de previsão e todas as bounding boxes do ground-truth. O maior valor de IoU encontrado é então utilizado como critério para associar a previsão à nuvem correspondente, ou seja, a dupla de boun-

ding boxes com o maior IoU é considerada como a correspondência mais próxima (ou a mais correta)[1]. Os valores de IoU são valores reais entre 0 e 1.

4.2 Aplicação dos Modelos

Para testar os modelos, foram escolhidas 4 imagens que não participaram dos treinamentos. Associada a cada imagem, existe uma lista de valores obtidos com o algoritmo do IoU, que representam as previsões significativas, ou seja, que tiveram um valor acima do **limiar de 0.5**.

As imagens abaixo contém os *bounding boxes* das **previsões em verde**, e do **gabarito em amarelo**.

4.2.1 Modelo 1

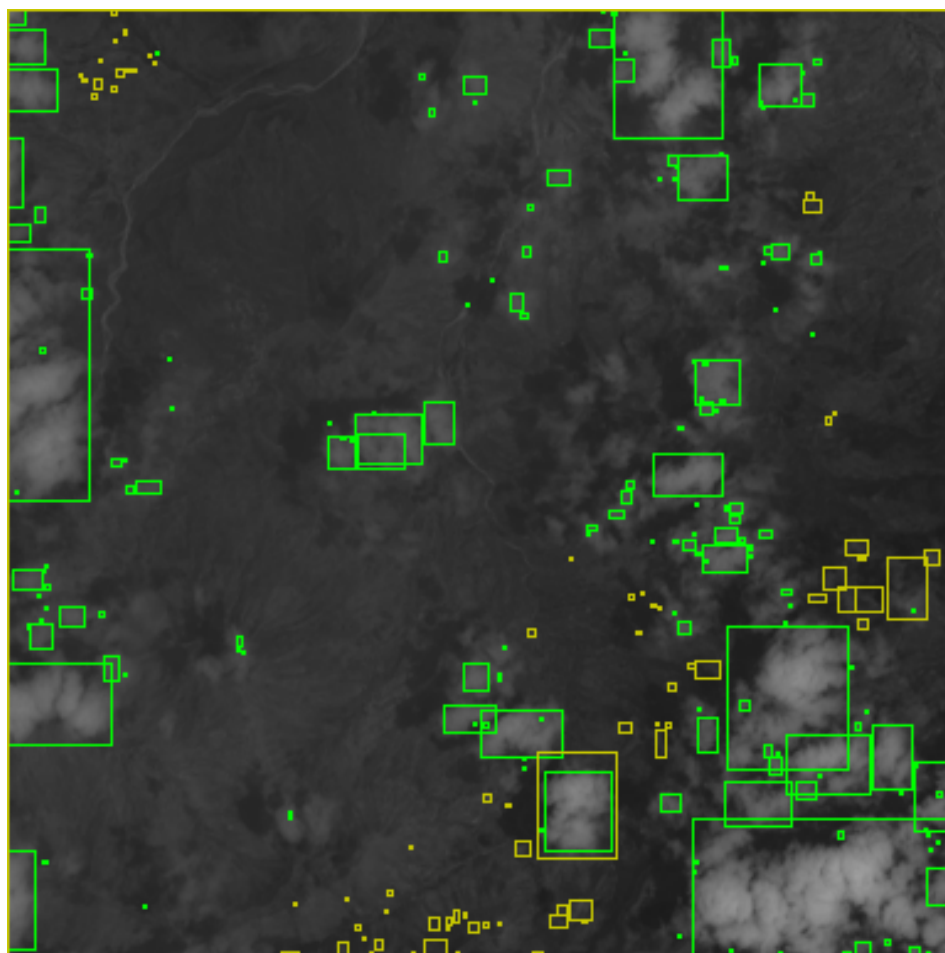


Figura 1: Imagem 1

IoUs: [0.5]

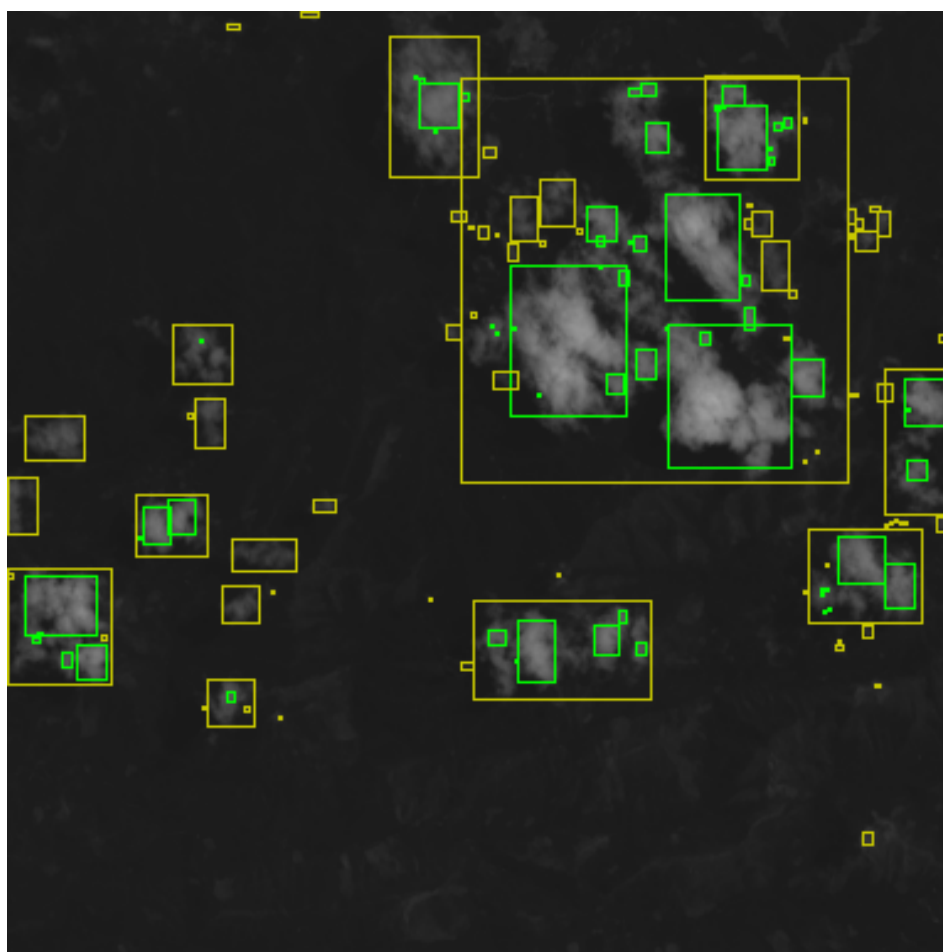


Figura 2: Imagem 2

IoUs: Sem IoUs significativos

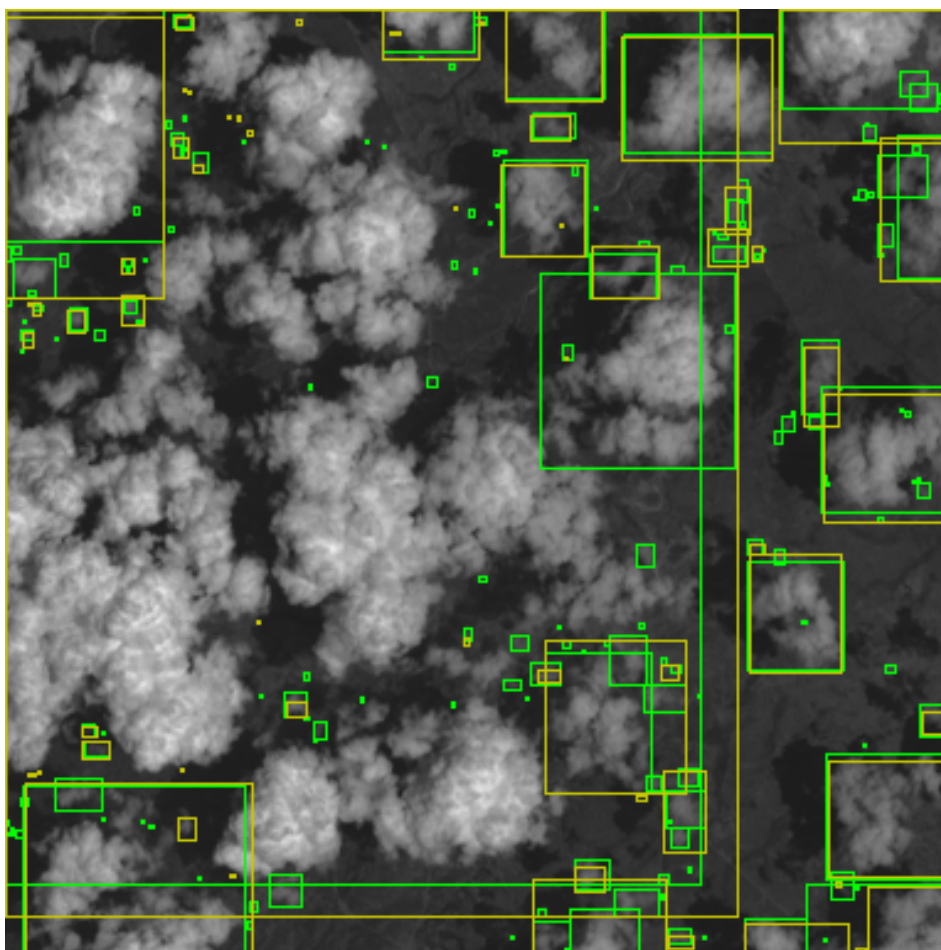


Figura 3: Imagem 3

$$\text{IoUs:} \begin{bmatrix} 1.0 & 0.6 & 0.8 & 0.8 & 0.5 & 0.5 & 0.6 & 0.7 & 0.8 & 0.6 \\ 0.6 & 0.6 & 0.5 & 0.7 & 0.9 & 0.5 & 0.7 & 0.9 & & \end{bmatrix}$$

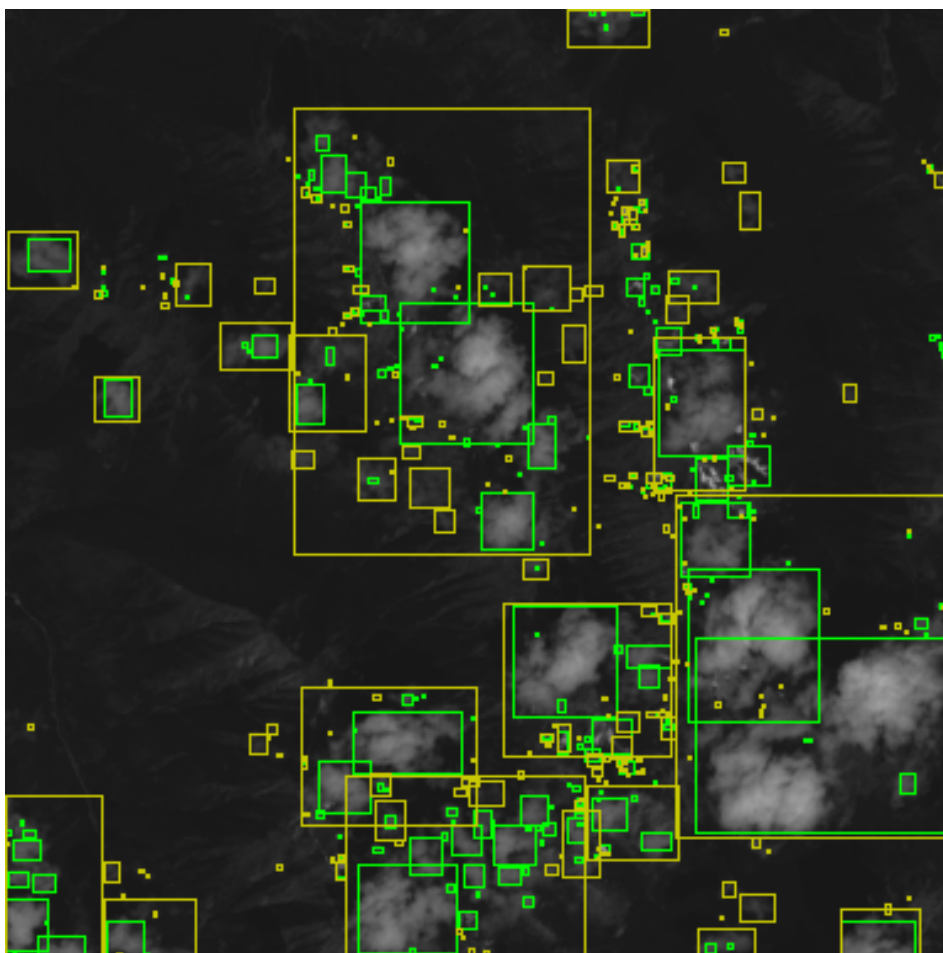


Figura 4: Imagem 4

$$\text{IoUs:} \begin{bmatrix} 0.5 & 0.8 & 1.0 & 0.5 & 1.0 & 1.0 & 1.0 & 0.6 & 0.5 & 0.5 \\ 1.0 & 0.5 & & & & & & & & \end{bmatrix}$$

4.2.2 Modelo 2

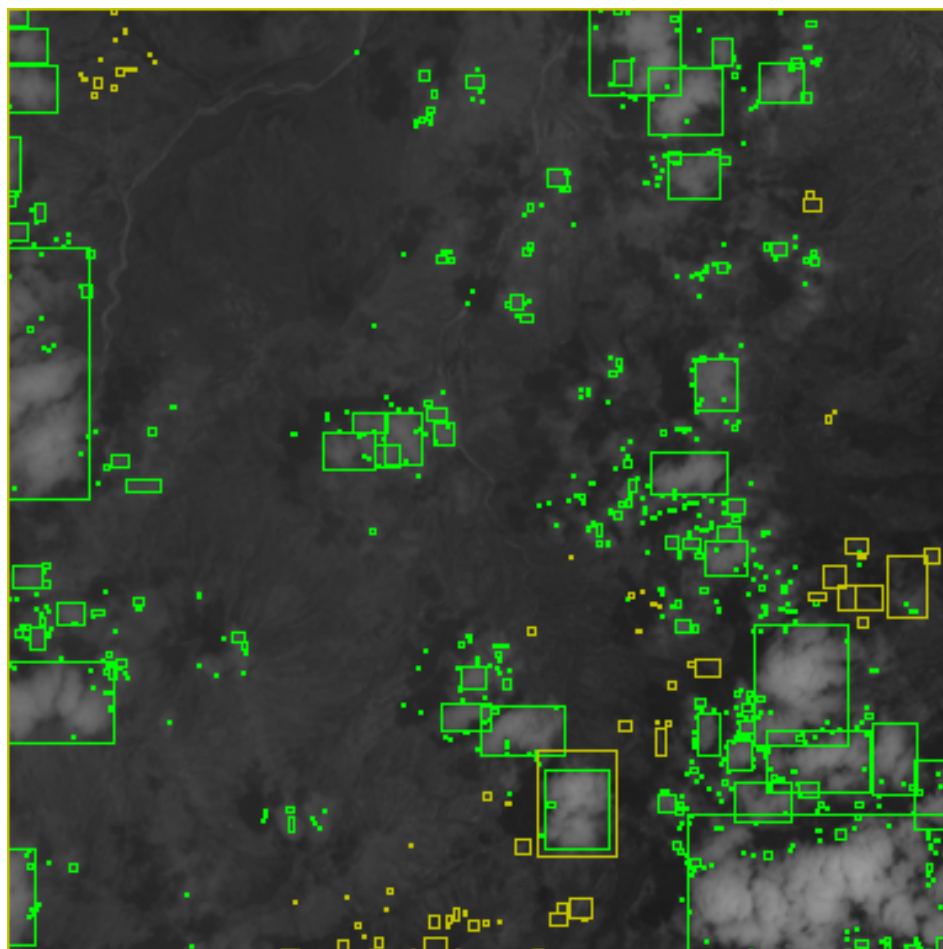


Figura 5: Imagem 1

IoUs: [0.5]

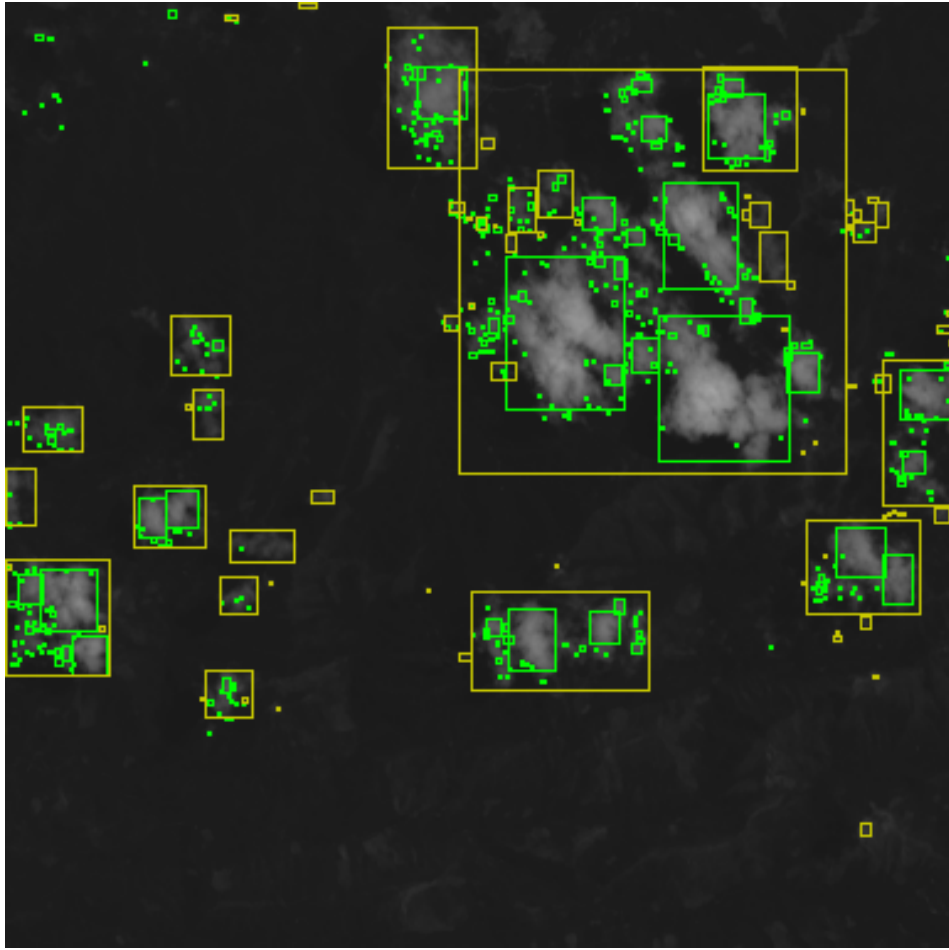


Figura 6: Imagem 2

IoUs: [1.0]

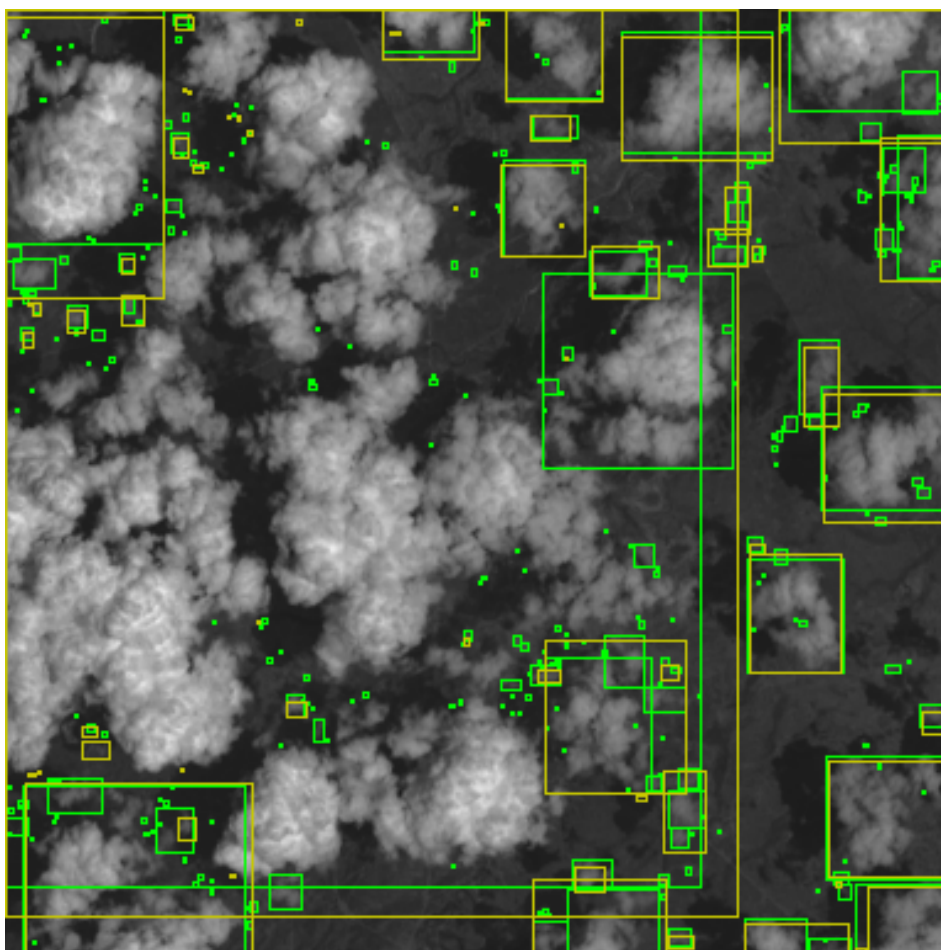


Figura 7: Imagem 3

$$\text{IoUs:} \begin{bmatrix} 0.6 & 0.6 & 0.8 & 0.5 & 0.6 & 0.7 & 0.8 & 0.6 & 0.7 & 0.8 \\ 0.5 & 0.6 & 0.5 & 0.5 & 0.8 & 0.6 & 0.9 & 0.7 & 0.9 & \end{bmatrix}$$

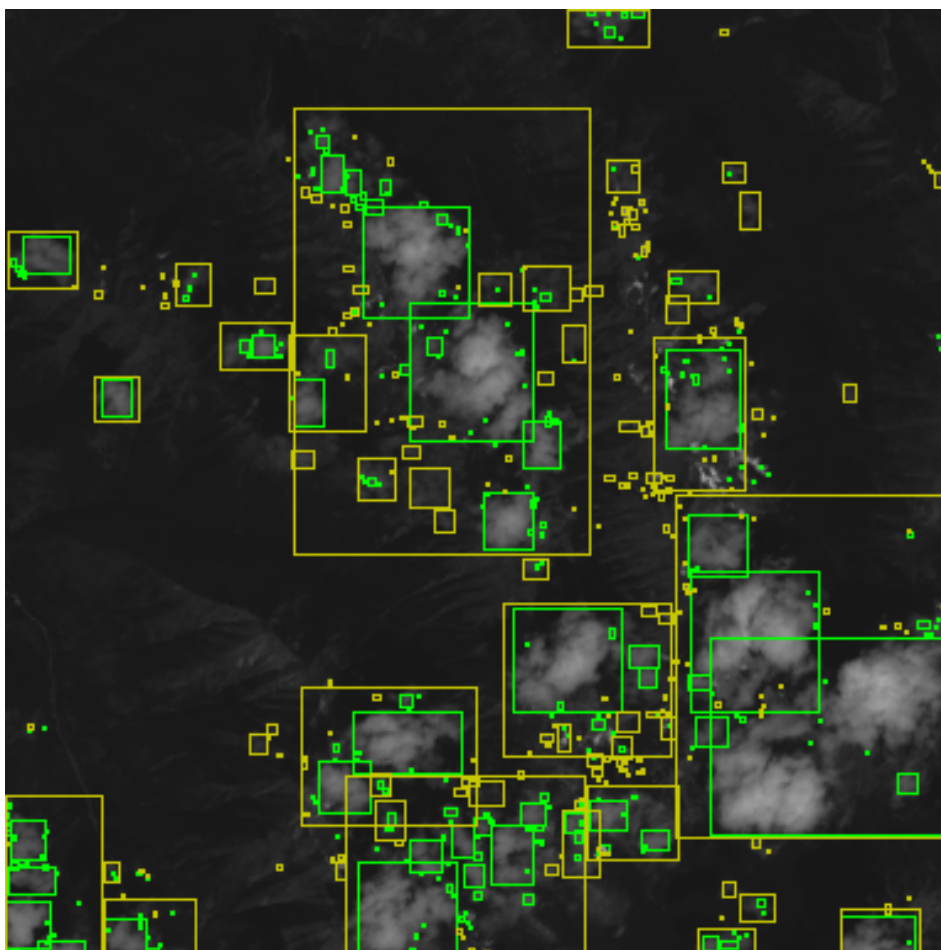


Figura 8: Imagem 4

IoUs: $[0.7 \quad 0.5 \quad 1.0]$

4.2.3 Modelo 3

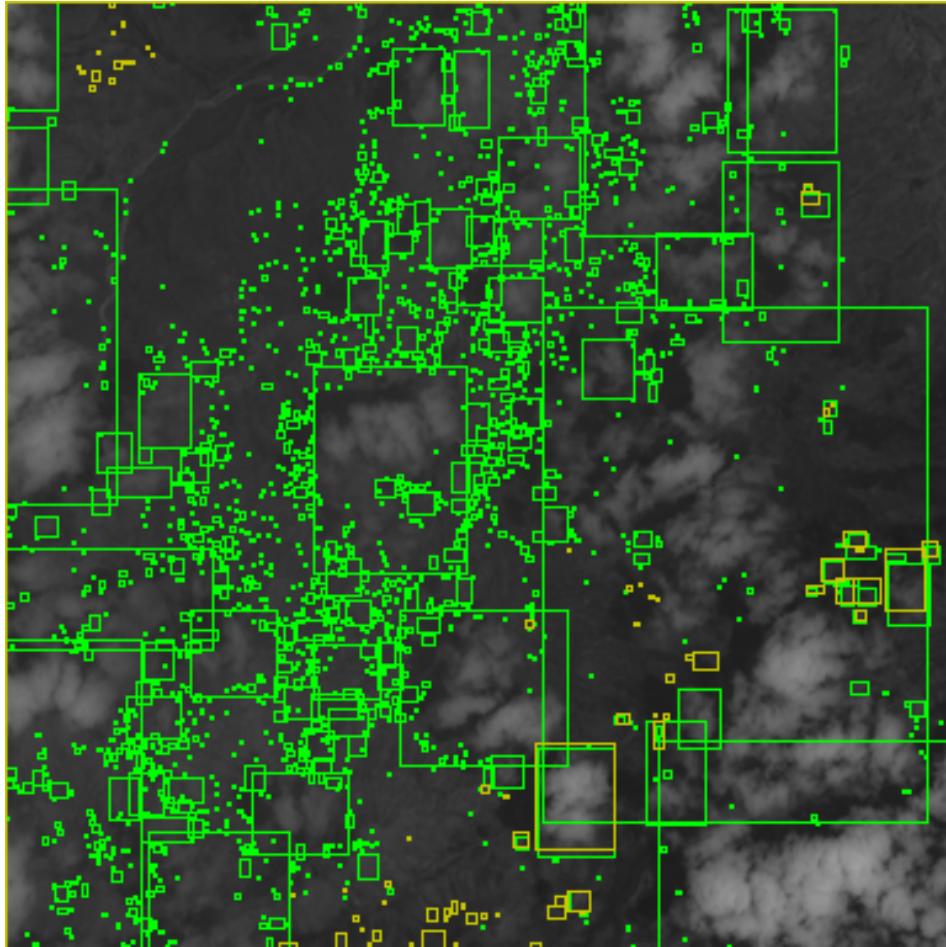


Figura 9: Imagem 1

IoUs: [0.7 0.5 0.8 0.6 0.6 0.7 0.5]

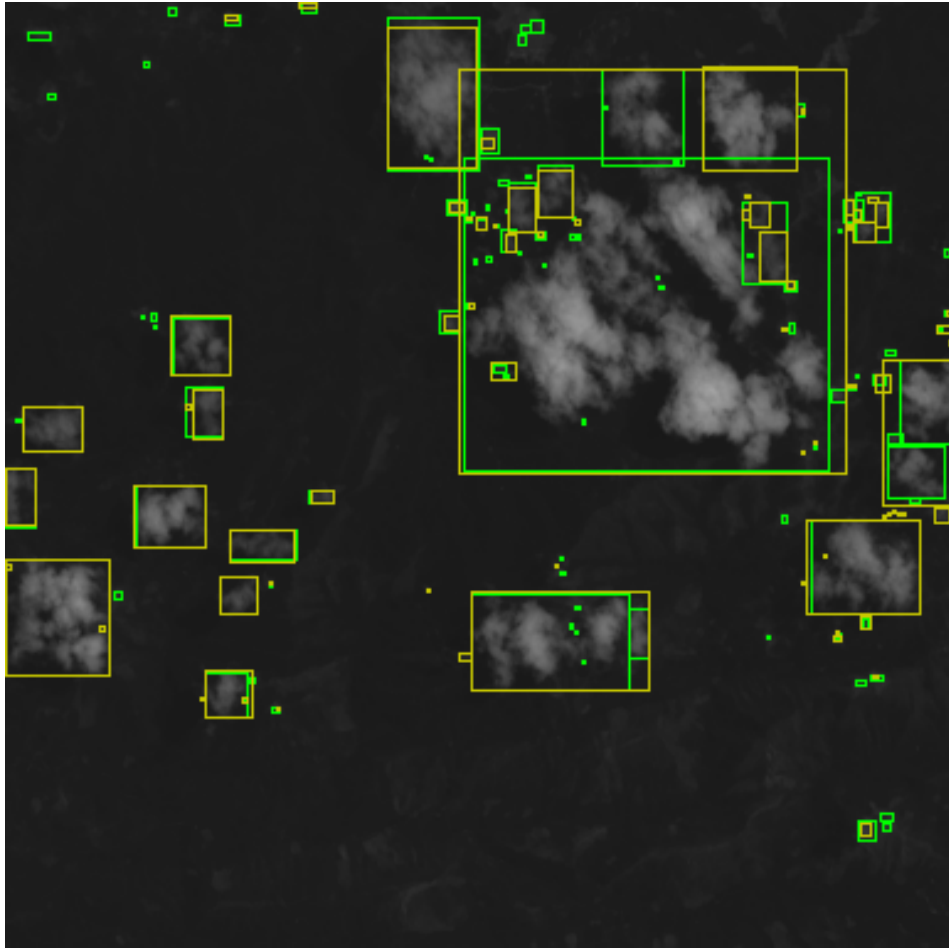


Figura 10: Imagem 2

$$\text{IoUs: } \begin{bmatrix} 0.8 & 0.8 & 1.0 & 1.0 & 1.0 & 0.8 & 0.8 & 1.0 & 0.6 & 0.8 \\ 0.9 & 1.0 & 0.6 & 0.9 & 0.5 & 0.5 & 0.5 & 0.5 & 0.6 & 0.5 \\ 1.0 & 0.6 & 0.7 & 1.0 & 0.7 & 0.5 & & & & \end{bmatrix}$$

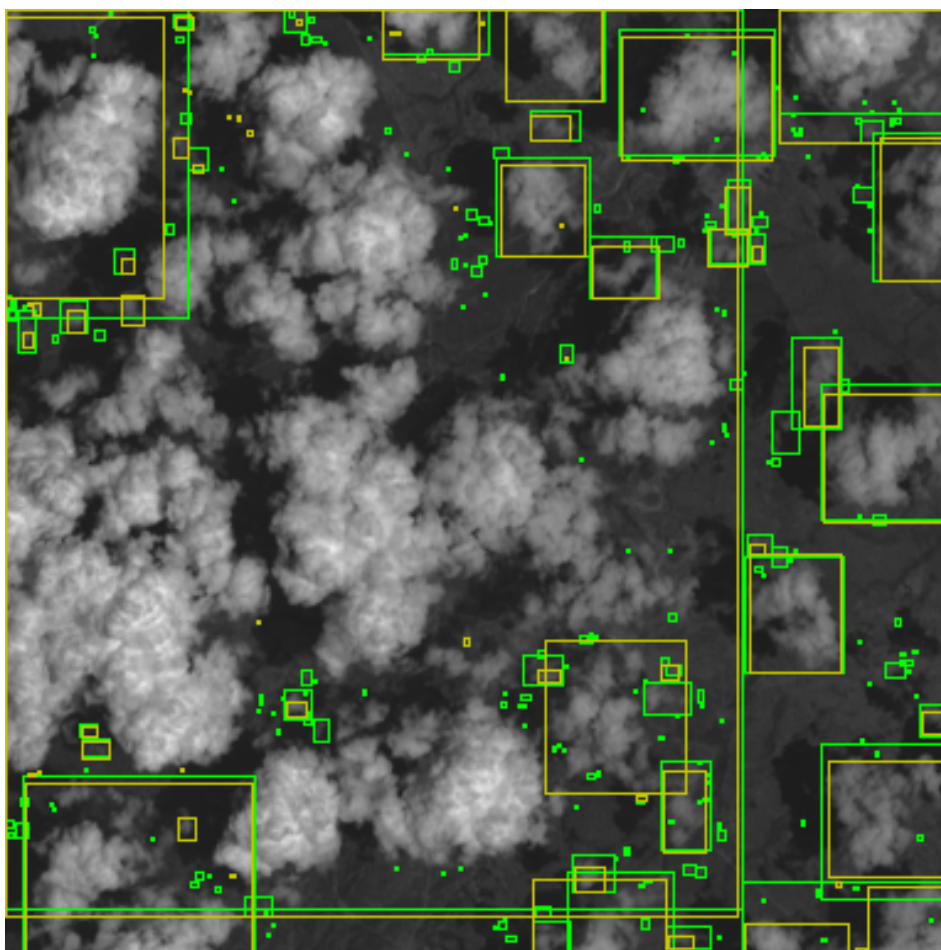


Figura 11: Imagem 3

$$\text{IoUs:} \begin{bmatrix} 1.0 & 0.7 & 0.6 & 0.8 & 0.7 & 0.5 & 0.6 & 0.7 & 0.8 & 0.6 \\ 0.6 & 0.8 & 0.8 & 0.8 & 1.0 & 0.6 & 0.9 & 0.5 & 0.7 & 0.9 \end{bmatrix}$$

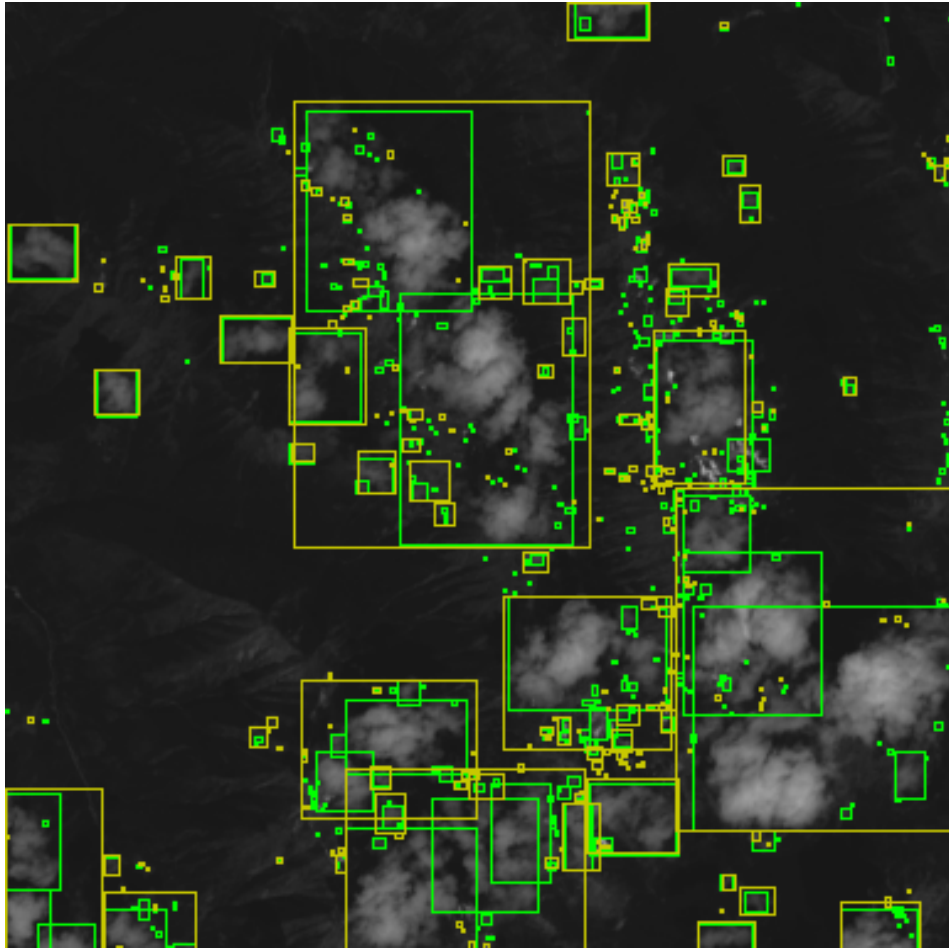


Figura 12: Imagem 4

$$\text{IoUs:} \begin{bmatrix} 0.8 & 0.7 & 0.7 & 0.5 & 0.6 & 0.7 & 1.0 & 0.7 & 0.9 & 0.7 \\ 0.6 & 1.0 & 1.0 & 0.6 & 0.6 & 1.0 & 1.0 & 0.7 & 0.8 & 1.0 \\ 0.5 & 0.7 & 0.7 & 1.0 & 0.9 & 0.6 & 1.0 & 0.5 & 1.0 & 0.8 \\ 0.6 & 0.7 & 0.7 & 1.0 & 0.5 & 0.6 & & & & \end{bmatrix}$$

4.2.4 Modelo 4

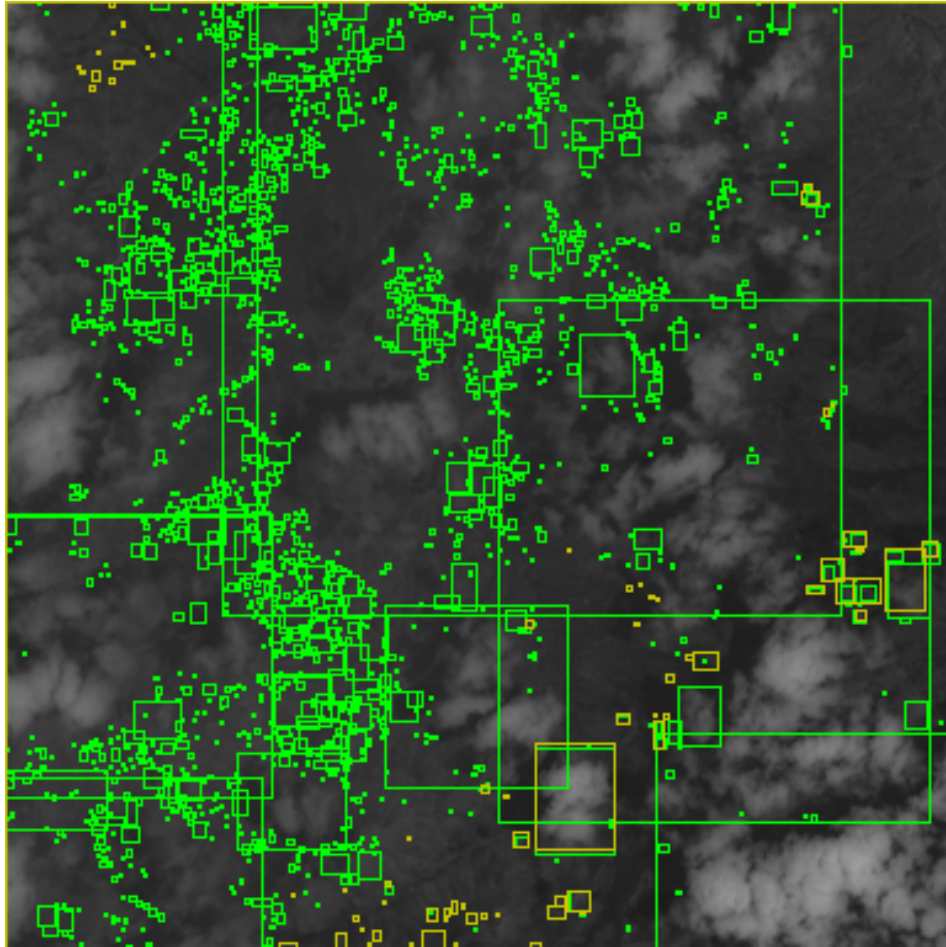


Figura 13: Imagem 1

IoUs: [0.6 0.7 0.8 0.5 0.6 0.5 0.5 0.5 0.7 0.6]

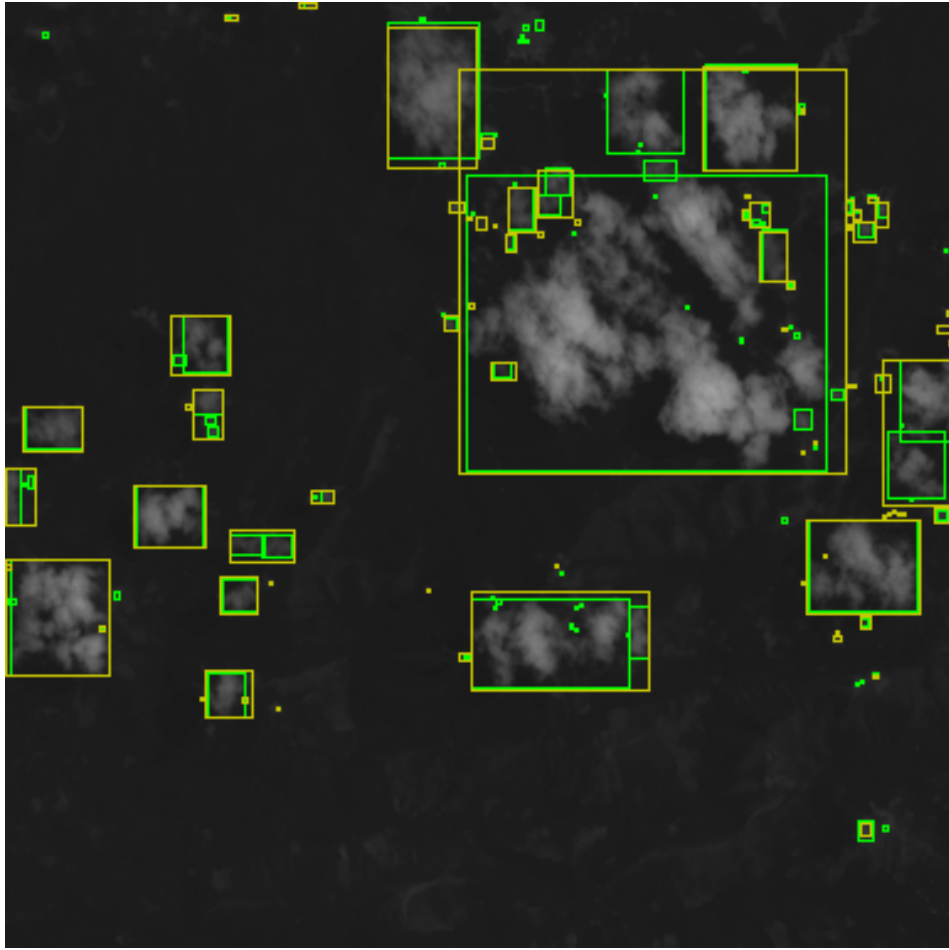


Figura 14: Imagem 2

$$\text{IoUs: } \begin{bmatrix} 1.0 & 0.5 & 0.8 & 0.8 & 0.9 & 0.5 & 0.8 & 0.5 & 0.9 & 0.5 \\ 0.6 & 0.7 & 0.5 & 0.7 & 1.0 & 0.6 & 0.5 & 0.5 & 0.5 & 1.0 \\ 0.6 & 0.8 & 0.8 & 0.8 & & & & & & \end{bmatrix}$$

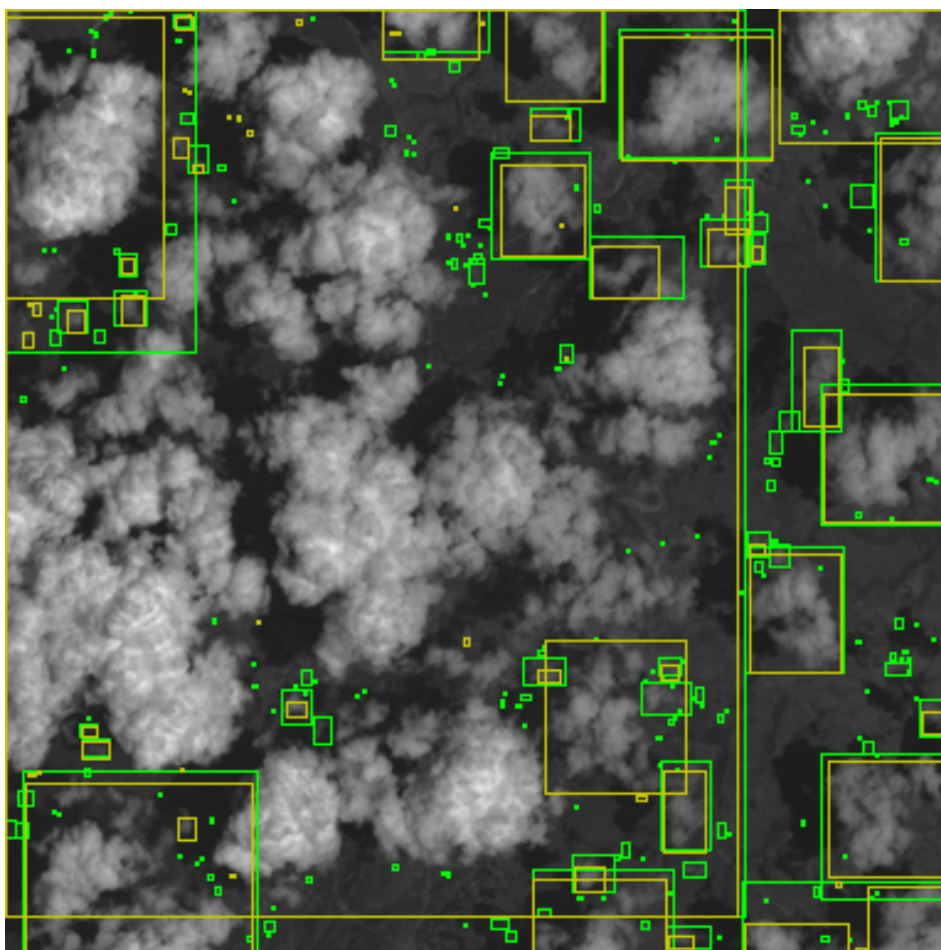


Figura 15: Imagem 3

$$\text{IoUs: } \begin{bmatrix} 0.6 & 0.8 & 0.7 & 0.7 & 0.8 & 0.7 & 0.5 & 0.8 & 0.8 & 0.5 \\ 0.5 & 0.5 & 0.6 & 0.6 & 0.5 & 0.9 & 0.6 & 0.8 & 0.5 & 1.0 \\ 0.9 \end{bmatrix}$$

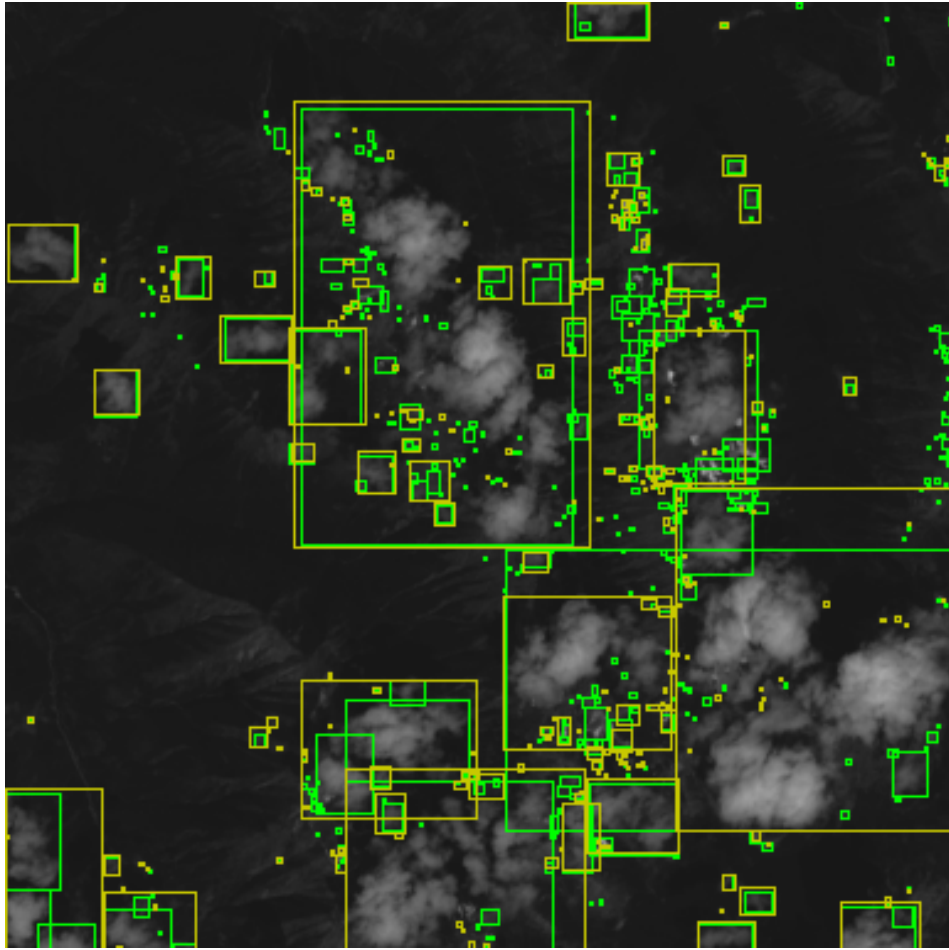


Figura 16: Imagem 4

$$\text{IoUs:} \begin{bmatrix} 0.8 & 0.8 & 0.6 & 0.8 & 0.6 & 1.0 & 0.6 & 0.6 & 0.6 & 1.0 \\ 0.7 & 0.8 & 0.7 & 0.6 & 1.0 & 0.5 & 1.0 & 0.8 & 0.6 & 1.0 \\ 0.6 & 0.5 & 0.5 & 0.8 & 0.6 & 1.0 & 0.8 & 0.9 & 0.6 & 0.8 \\ 0.8 & 0.6 & 1.0 & 0.9 & 1.0 & 1.0 & 0.7 & 0.6 & & \end{bmatrix}$$

5 Conclusão

No presente trabalho, foram investigados métodos de detecção de nuvens com base no conjunto de dados **38-Cloud**, que oferece imagens de satélite multiespectrais e suas correspondentes máscaras de segmentação de nuvens. Utilizando técnicas de aprendizado de máquina e análise de imagens, foram elaborados quatro modelos de detecção de nuvens, cujo desempenho progrediu à medida que o volume de alimentação e balanceamento dos pixels aumentaram (Modelos 1, 2, 3 e 4).

Os resultados experimentais evidenciaram que os modelos foram capazes de identificar com certa precisão as nuvens nas imagens de teste, gerando previsões robustas em uma diversidade de condições atmosféricas. Contudo, notaram-se limitações na detecção de nuvens em cenários específicos, como nuvens finas, fragmentadas ou parcialmente obscurecidas por outras características da cena.

Ao prever futuros trabalhos, deverá ser considerada a inclusão da alimentação de imagens com gelo e neve, ampliando ainda mais a aplicabilidade e a precisão dos modelos desenvolvidos.

Referências

- [1] AruniRC. Python — sentiment analysis using vader. Gist. <https://gist.github.com/AruniRC/c629c2df0e68e23aff7dcaeef87c72d4>.
- [2] S. Mohajerani, T. A. Krammer, and P. Saeedi. A Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–5, Aug 2018.
- [3] S. Mohajerani and P. Saeedi. Cloud-Net: An End-To-End Cloud Detection Algorithm for Landsat 8 Imagery. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 1029–1032, 2019.
- [4] PyCaret. Tutorial - binary classification. GitHub, s.d. <https://github.com/pycaret/pycaret/blob/master/tutorials/Tutorial%20-%20Binary%20Classification.ipynb>.