

**Deep Learning aplicado na
classificação de imagens de
satélite**

**Aquisição de imagens
de sensores de satélite**

Priscila M. Kai



Roteiro

01

Aula 1

Introdução ao
Sensoriamento Remoto

02

Aula 2

*Aquisição de imagens de
sensores de satélite*

03

Aula 3

Extração de
características

04

Aula 4

Construção de Modelos
para a Classificação de
Culturas

05

Aula 5

Construção de uma Rede
Neural Densa



02

Aquisição de Imagens de Sensores de satélite

O que veremos?

Roteiro

1. Imagens de sensores de satélite
 - Imagem Hiperespectral
 - Imagem Multiespectral
2. Plataformas online
 - SentinelHub
 - Combinações úteis para a agricultura
 - EOS LandViewer
3. Manipulação de imagens do Sentinel-2 com Python
 - Carregando imagens
 - Selecionando regiões de interesse

Imagem Multiespectral e Hiperespectral

Imagens **multiespectrais** e **hiperespectrais** consistem em várias bandas de dados.

As bandas da imagem podem ser exibidas por vez, apresentadas por imagem em escala de cinza, ou em combinação de três bandas por vez como uma imagem composta colorida.

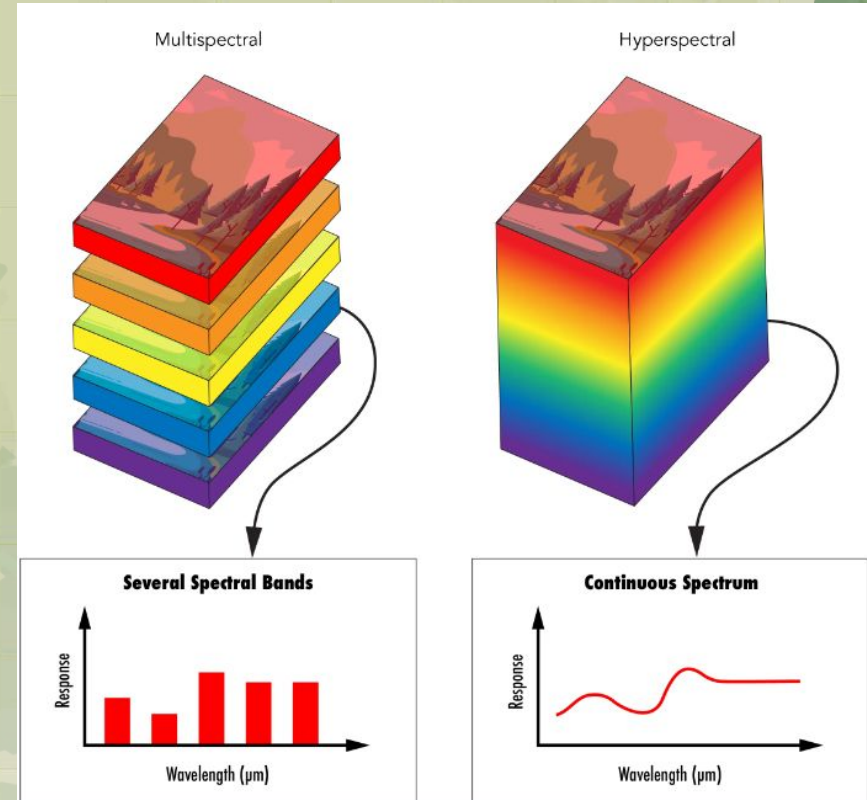


Imagem Multiespectral e Hiperespectral

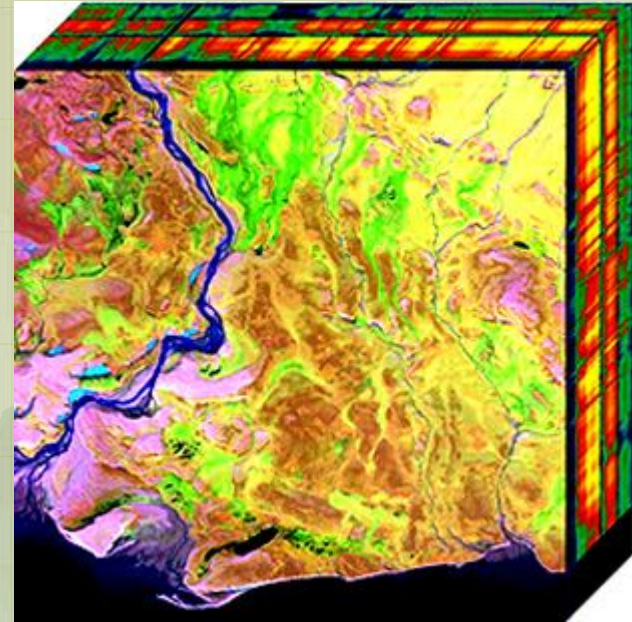
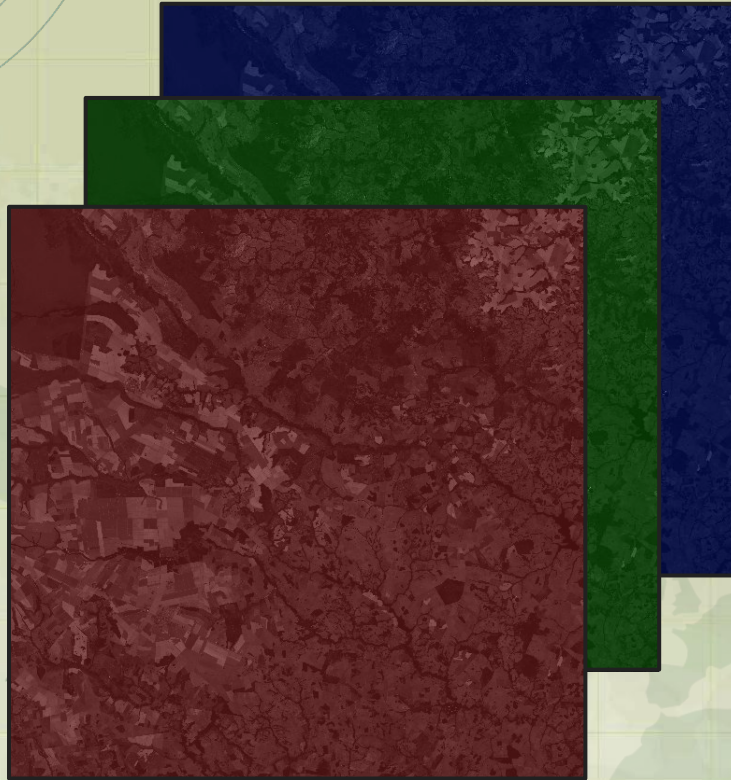
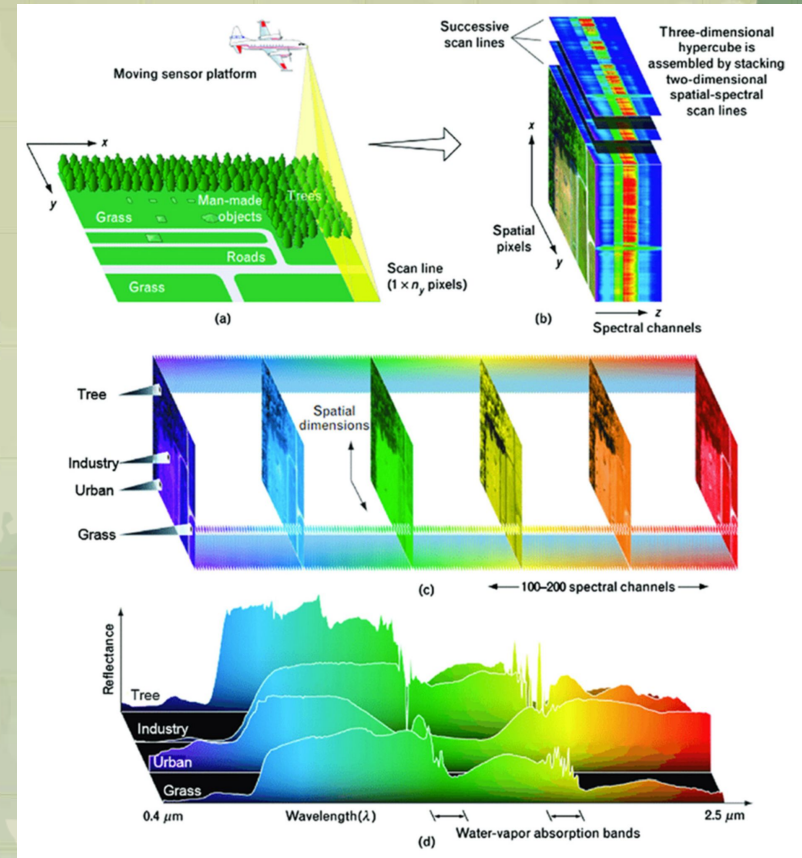


Imagem Multiespectral e Hiperespectral

A imagem **hiperespectral** possui a coleta de centenas de imagens em diferentes comprimentos de onda para a mesma área.



Plataformas online

Para a coleta de imagens de produtos do sensoriamento remoto



Sentinel Hub Playground

<https://www.sentinel-hub.com/explore/sentinelplayground/>

The screenshot displays the Sentinel Hub Playground web interface. The top navigation bar includes a back arrow, the Sentinel Hub logo, the text "Playground", a calendar icon showing the date "2022-10-19", a cloud icon with "92.22 %", a search icon with the text "Go to Place", and zoom controls (+, -, and a crosshair). Below the navigation bar, the left sidebar contains a "Rendering" section with a list of preset styles: "Custom", "Natural color (Based on bands 4,3,2)", "Color Infrared (vegetation) (Based on bands 8,4,3)", "Vegetation Index (Based on combination of bands (B8 - B4)/(B8 + B4))", "False color (urban) (Based on bands 12,11,4)", "Moisture index (Based on combination of bands (B8A - B11)/(B8A + B11))", "SWIR (Based on bands 12,8A,4)", "NDWI (Based on combination of bands (B3 - B6)/(B3 + B6))", and "NDSI (Based on combination of bands (B3 - B6)/(B3 + B6))". A "GENERATE" button is located at the bottom of this list. The main area shows a satellite image of a city, with a zoom slider positioned above it. A scale bar in the bottom right corner indicates "500 m".

Combinações de bandas úteis para a agricultura

Além dos índices de vegetação podemos obter mais características de cada classe através de **combinações RGB**. Ao combinarmos três bandas, temos uma imagem colorida com diferentes proporções de vermelho, verde e azul para cada pixel da imagem de saída.

Uma imagem RGB natural é composta pela banda do **vermelho**, **verde** e **azul** do espectro visível. Para uma imagem em cor falsa, temos rearranjos de bandas, incluindo comprimentos de onda não visíveis a fim de revelar informações não disponíveis na representação RGB tradicional.



Combinações de bandas úteis para a agricultura

- ❑ Color Infrared (B8, B4, B3)
- ❑ Short-Wave Infrared (B12, B8A, B4)
- ❑ Agriculture (B11, B8, B2)

Combinações de bandas úteis para a agricultura

- **Color Infrared (B8, B4, B3)**
 - combinação de cores por uso de bandas do **infravermelho próximo (B8)**, **vermelho (B4)** e **verde (B3)**. Avalia a densidade e saúde de plantas. Folhagem densa é representada em vermelho escuro, enquanto solo exposto, regiões urbanas são coloridas com cinza ou marrom e água em azul escuro ou preto.
- Short-Wave Infrared (B12, B8A, B4)
- Agriculture (B11, B8, B2)



Combinações de bandas úteis para a agricultura

- Color Infrared (B8, B4, B3)
- **Short-Wave Infrared (B12,B8A,B4)**
 - combinação de cores por uso de bandas do **infravermelho de ondas curtas (B12)**, **infravermelho próximo (B8A)** e **vermelho (B4)**. A banda do infravermelho próximo B8A reflete a vegetação, apresentando regiões com cores em verde claro, solo nu e construções em marrom ou azulado e água em preto.
- Agriculture (B11, B8, B2)



Combinações de bandas úteis para a agricultura

- Color Infrared (B8, B4, B3)
- Short-Wave Infrared - SWIR (B12, B8A, B4)
- **Agriculture (B11, B8, B2)**
 - Combinação de cores com as bandas do **infravermelho de ondas curtas (B11)**, **infravermelho próximo (B8)** e **azul (B02)**. Usado no monitoramento da saúde das plantas. Bandas do infravermelho de ondas curtas como a B11 e B12 auxiliam na estimação da quantidade de água em plantas e no solo, refletida na luz SWIR.



Combinações de bandas úteis para a agricultura



EOS LandViewer

The screenshot displays the EOS LandViewer web application interface. The main map shows a region in Goiás, Brazil, with various cities and towns labeled, including Anápolis, Goiânia, Trindade, and Aparecida de Goiânia. The map is overlaid with a grid. On the left side, there is a vertical toolbar with icons for map navigation and analysis. At the top left, there is a search bar with the text "Locais, coordenadas, ID da cena" and a magnifying glass icon. Below the search bar, there is a button labeled "Upload de Área de Interesse (AOI)".

On the right side, there is a panel titled "PESQUISA DE CENA" (Scene Search). This panel contains a section for "Configurações de busca" (Search Settings) with a toggle for "Sensores passivos (dia)" (Passive sensors (day)) and a "Mosaico" (Mosaic) option. Below this, there is a list of search results, each showing a thumbnail image, the date, and the sensor used. The results are as follows:

Thumbnail	Date	Sensor	Resolution	Clouds	Count
	19 Out 2022	Sentinel-2 L2A	68"	2%	22KFG
	17 Out 2022	Landsat 8 RT	64"	1%	222/071
	17 Out 2022	Landsat 8 RT	64"	2%	

At the bottom of the search results, there is a button labeled "Comprar imagens de alta resolução" (Buy high-resolution images). The bottom status bar shows the coordinates "16.19226°S 49.34509°W" and a scale of "10 km".

Manipulação de imagens do Sentinel-2 com Python

Para realizarmos a classificação entre diferentes classes, primeiramente é necessário obter um conjunto de dados (dataset), ou realizar o download de imagens e então selecionar as regiões de estudo por meio das coordenadas do local.

Caso a segunda opção seja escolhida, podemos realizar o download de imagens de cada local através de um arquivo shapefile, ou com extensão .kml (*Keyhole Markup Language*) com o **Google Earth**.

Manipulação de imagens do Sentinel-2 com Python

Selecionando regiões com o **Google Earth**

- Primeiramente é necessário criar um novo projeto, selecionando locais a partir de linhas que demarcam a região a ser estudada, criando polígonos.
- Selecionados os polígonos, podemos fazer o download de cada forma criada em um arquivo .kml.
- O arquivo servirá para recortarmos as regiões de interesse das imagens baixadas da plataforma online.



Manipulação de imagens do Sentinel-2 com Python

Para a manipulação de imagens do Sentinel-2 será utilizado o **Spyder**, ambiente de desenvolvimento Python.

O tutorial de instalação do Anaconda com Spyder pode ser encontrado no seguinte [link](#).

Manipulação de imagens do Sentinel-2 com Python

Selecionando regiões com o **Google Earth**

- Primeiramente é necessário criar um novo projeto, selecionando locais a partir de linhas que demarcam a região a ser estudada, criando polígonos.
- Selecionados os polígonos, podemos fazer o download de cada forma criada em um arquivo .kml.
- O arquivo servirá para recortarmos as regiões de interesse das imagens baixadas da plataforma online.



Carregando imagens do Sentinel-2

Importando bibliotecas:

```
import geopandas as gpd  
import matplotlib.pyplot as plt  
import rasterio  
from os import listdir  
from os.path import isfile, join  
from rasterio.mask import mask
```

Carregando imagens do Sentinel-2

Lendo arquivo .kml:

```
# Lendo KML files com GeoPandas
gpd.io.file.fiona.drvsupport.supported_drivers['KML'] = 'rw'
my_map = gpd.read_file('Arquivo.kml', driver='KML')

# Convertendo EPSG para o mesmo da imagem do Sentinel-2
my_map = my_map.to_crs(epsg=32722)

# Pasta contendo imagens
path_input = 'caminho_contendo_imagens_de_entrada'

# Obtendo nomes de arquivos de imagem
img_nomes = [f for f in listdir(path_input) if isfile(join(path_input, f))]
full_path = join(path_input, img_nomes[3])
```


Selecionando regiões de interesse

Recortando imagens:

```
#Recortando imagens
with rasterio.open(full_path) as src:
    out_image, out_transform = mask(src, [my_map.geometry[0]], crop=True)
    out_meta = src.meta.copy()
    out_image = out_image.squeeze()
```