

GeoAI mit Python

*Ein praxisorientierter Leitfaden
für Open-Source-Geodaten-KI*



Qiusheng Wu

GeoAI mit Python

Ein praxisorientierter Leitfaden für Open-Source-
Geodaten-KI

Qiusheng Wu
2026

Contents

Vorwort	1
Einführung	3
Für wen dieses Buch ist	4
Was dieses Buch behandelt	5
Das Beste aus diesem Buch herausholen	8
In diesem Buch verwendete Konventionen	9
Herunterladen der Codebeispiele	10
Video-Tutorials	10
Community und Feedback	11
Über den Autor	11
Lizenzierung und Urheberrecht	12
I: Grundlagen	13
1. Einführung in GeoAI	15
1.1. Einleitung	15
1.2. Lernziele	16
1.3. Was ist GeoAI?	16
1.4. Zentrale Anwendungsgebiete von GeoAI	18
1.5. Das Python-Ökosystem für GeoAI	22
1.6. Grundlegende KI-Aufgaben in der Geodatenanalyse	25
1.7. Von traditionellen Methoden zu Foundation Models	32
1.8. Zentrale Erkenntnisse	33
1.9. Übungen	34
2. Einrichtung Ihrer Umgebung	37
2.1. Einführung	37
2.2. Lernziele	37
2.3. Hardwareanforderungen	37
2.4. Installation der NVIDIA-Treiber	38
2.5. Installation von Python mit Miniconda	41
2.6. Erstellen einer Conda-Umgebung	43
2.7. Installation von GeoAI und PyTorch mit GPU-Unterstützung	43
2.8. Installation von uv als alternativer Paketmanager	45
2.9. Einrichten von Jupyter	46
2.10. Installation von Visual Studio Code	47
2.11. Überprüfung Ihres Setups	48
2.12. Übungen	50
3. Grundlagen Geodaten	51
3.1. Einleitung	51
3.2. Lernziele	51
3.3. Rasterdaten	51
3.4. Vektordaten	55
3.5. Koordinatenreferenzsysteme	58
3.6. Annotationsformate für Deep Learning	59
3.7. Bildkachelung und Chips	61
3.8. Zusammenfassung	62
3.9. Übungen	62

II: Datenerfassung und -aufbereitung	65
4. Herunterladen von Fernerkundungsdaten	67
4.1. Einführung	67
4.2. Lernziele	67
4.3. Satellitenbildquellen	68
4.4. Suchen mit STAC	69
4.5. Herunterladen von NAIP-Bildern	75
4.6. Herunterladen von Sentinel-2-Daten	76
4.7. Herunterladen von Landsat-Daten	77
4.8. Herunterladen von Vantor Open Data	79
4.9. Zugriff auf Vektordaten	80
4.10. Organisation Ihrer Daten	82
4.11. Wichtige Erkenntnisse	83
4.12. Übungen	84
5. Interaktive Kartierung und Visualisierung	87
5.1. Einleitung	87
5.2. Lernziele	87
5.3. Interaktive Karten mit Leafmap erstellen	88
5.4. Arbeiten mit Rasterdaten auf Karten	89
5.5. Visualisierung von Daten vom Planetary Computer	92
5.6. Arbeiten mit Vektordaten auf Karten	96
5.7. Split-Panel-Karten zum Vergleich	98
5.8. Visualisierung von Modellergebnissen	99
5.9. Best Practices fuer GeoAI-Visualisierung	101
5.10. Wichtigste Erkenntnisse	102
5.11. Uebungen	103
6. Erstellen von Trainingsdaten	105
6.1. Einführung	105
6.2. Lernziele	105
6.3. Die Trainingsdaten-Pipeline	105
6.4. Erzeugen von Bildkacheln aus einem Einzelbild	106
6.5. Stapelverarbeitung mehrerer Bilder	111
6.6. Stapelverarbeitung mit Rastermasken	115
6.7. Überlegungen zur Labelqualität	116
6.8. Organisation des Datensatzes	116
6.9. Zusammenfassung	118
6.10. Wichtige Erkenntnisse	118
6.11. Übungen	119
III: KI-Kernaufgaben	121
7. Bilderkennung	123
7.1. Einführung	123
7.2. Lernziele	123
7.3. Bildklassifikation verstehen	124
7.4. Klassifikationsarchitekturen	125
7.5. Daten für die Klassifikation vorbereiten	126
7.6. Bibliotheken importieren	127
7.7. Den EuroSAT-RGB-Datensatz herunterladen	128
7.8. Den Datensatz erkunden	128

7.9. Einen ResNet-50-Klassifikator trainieren	130
7.10. Trainingsverlauf darstellen	131
7.11. Auf dem Testset evaluieren	131
7.12. Konfusionsmatrix darstellen	132
7.13. Vorhersagen visualisieren	134
7.14. Einen EfficientNet-B0-Klassifikator trainieren	135
7.15. Ergebnisse vergleichen	137
7.16. Modelle veröffentlichen und wiederverwenden	137
7.17. Wichtige Erkenntnisse	139
7.18. Übungen	140
8. Objekterkennung	141
8.1. Einführung	141
8.2. Lernziele	141
8.3. Objekterkennung verstehen	142
8.4. Erkennungsarchitekturen	143
8.5. Vorbereitung von Erkennungsdatensätzen	146
8.6. Bewertung von Erkennungsergebnissen	146
8.7. Bibliotheken importieren	148
8.8. Den NWPU-VHR-10-Datensatz herunterladen	148
8.9. Den Datensatz erkunden	148
8.10. Den Datensatz vorbereiten	149
8.11. Beispielannotationen visualisieren	149
8.12. Ein mehrklassiges Erkennungsmodell trainieren	150
8.13. Trainingsmetriken visualisieren	151
8.14. Bewertung mit COCO-Metriken	152
8.15. Inferenz auf Beispielf Bildern durchführen	152
8.16. Erkennungen visualisieren	153
8.17. Batch-Inferenz auf mehreren Bildern	154
8.18. Modelle veröffentlichen und wiederverwenden	156
8.19. Wichtigste Erkenntnisse	158
8.20. Übungen	159
9. Semantische Segmentierung	161
9.1. Einführung	161
9.2. Lernziele	161
9.3. Grundlagen der semantischen Segmentierung	162
9.4. Bibliotheken importieren	164
9.5. Gebäudeerfassung aus Luftbildern	164
9.6. Oberflächenwasserkartierung	170
9.7. Wolken- und Wolkenschattenerkennung	194
9.8. Landbedeckungsklassifikation	201
9.9. Modelle veröffentlichen und wiederverwenden	205
9.10. Wichtige Erkenntnisse	206
9.11. Übungen	207
10. Instanzsegmentierung	209
10.1. Einleitung	209
10.2. Lernziele	209
10.3. Instanzsegmentierung vs. Semantische Segmentierung	210
10.4. Mask R-CNN-Architektur	211

10.5.	Herunterladen des FTW-Datensatzes	212
10.6.	Vorbereitung der Trainingsdaten	214
10.7.	Training eines Mask R-CNN-Modells	215
10.8.	Durchführung der Inferenz	216
10.9.	Nachbearbeitung der Vorhersagen	218
10.10.	Extraktion geometrischer Eigenschaften	221
10.11.	Stapelverarbeitung	223
10.12.	Wichtigste Erkenntnisse	223
10.13.	Übungen	224
11.	<i>Bildübersetzung</i>	227
11.1.	Einführung	227
11.2.	Lernziele	227
11.3.	Grundlagen der Bildübersetzung	228
11.4.	Bibliotheken importieren	230
11.5.	Beispieldaten herunterladen	230
11.6.	Visualisierung des RGB-Eingabekomposits	231
11.7.	Einzelpatch-Super-Resolution	232
11.8.	Unsicherheitsschätzung	234
11.9.	Gekachelte Inferenz für größere Regionen	235
11.10.	Einschränkungen und Vorsichtshinweise	237
11.11.	Wichtigste Erkenntnisse	238
11.12.	Übungen	239
12.	<i>Veränderungserkennung</i>	241
12.1.	Einführung	241
12.2.	Lernziele	241
12.3.	Veränderungserkennung verstehen	242
12.4.	Traditionelle Methoden der Veränderungserkennung	243
12.5.	Deep Learning für Veränderungserkennung	246
12.6.	Bibliotheken importieren	247
12.7.	Verfügbare Modelle auflisten	247
12.8.	Einrichtung	247
12.9.	Beispieldaten herunterladen	248
12.10.	Eingabebilder visualisieren	248
12.11.	ChangeStar-Modell initialisieren	248
12.12.	Veränderungserkennung ausführen	249
12.13.	Ergebnisse visualisieren	250
12.14.	Verwendung der Komfortfunktion	251
12.15.	Modellvarianten vergleichen	252
12.16.	Schwellenwert anpassen	253
12.17.	Gespeicherte Ausgaben anzeigen	253
12.18.	Wichtige Erkenntnisse	255
12.19.	Übungen	255
13.	<i>Pixelbasierte Regression</i>	257
13.1.	Einleitung	257
13.2.	Lernziele	257
13.3.	Pixelregression verstehen	258
13.4.	Regressionsarchitekturen	259
13.5.	Fallstudie: NDVI-Vorhersage aus Landsat-Bildern	260

13.6.	Evaluierungsmetriken	266
13.7.	Wichtigste Erkenntnisse	267
13.8.	Übungen	267
IV:	Foundation-Modelle und Satellitenembeddings	269
14.	<i>Segment Anything für Geodaten</i>	271
14.1.	Einführung	271
14.2.	Lernziele	272
14.3.	Funktionsweise von SAM 3	272
14.4.	Einrichten der Umgebung	273
14.5.	Bildsegmentierung	273
14.6.	Punkt-Prompts für Instanzsegmentierung	278
14.7.	Box-Prompts zur Gebäudeextraktion	285
14.8.	Stapelsegmentierung	289
14.9.	Gekachelte Segmentierung für große Bilder	291
14.10.	Interaktive Segmentierung	294
14.11.	Videosegmentierung	296
14.12.	Wichtigste Erkenntnisse	302
14.13.	Übungen	303
15.	<i>Vision-Language-Modelle</i>	305
15.1.	Einleitung	305
15.2.	Lernziele	305
15.3.	Funktionsweise von Vision-Language-Modellen	306
15.4.	Einrichten der Umgebung	307
15.5.	Beispieldaten	307
15.6.	Initialisierung des Moondream-Prozessors	308
15.7.	Bildbeschreibung	309
15.8.	Visuelle Fragenbeantwortung	310
15.9.	Objekterkennung und Punktlokalisierung	310
15.10.	Interaktive GUI	313
15.11.	Sliding-Window-Analyse fuer grosse Raster	315
15.12.	CLIP-basierte Segmentierung	322
15.13.	Praktische Anwendungen in der Erdbeobachtung	324
15.14.	Einschraenkungen und Ueberlegungen	325
15.15.	Wichtigste Erkenntnisse	325
15.16.	Uebungen	326
16.	<i>Satelliteneinbettungen</i>	329
16.1.	Einführung	329
16.2.	Lernziele	329
16.3.	Was sind Satelliten-Embeddings?	330
16.4.	Einrichten der Umgebung	330
16.5.	Verfügbare Embedding-Datensätze durchsuchen	331
16.6.	Patchbasierte Embeddings erkunden	332
16.7.	Embeddings visualisieren	334
16.8.	Embeddings clustern	336
16.9.	Ähnlichkeitssuche	338
16.10.	Klassifikatoren auf Embeddings trainieren	341
16.11.	Embeddings für die Veränderungserkennung vergleichen	345
16.12.	Datensätze mit TorchGeo laden	347

16.13.	Arbeiten mit temporalen TESSERA-Embeddings	347
16.14.	AlphaEarth-Satelliten-Embeddings erkunden	353
16.15.	Wichtige Erkenntnisse	356
16.16.	Übungen	357
V:	QGIS-Plugins	359
17.	<i>Einrichtung des GeoAI-QGIS-Plugins</i>	361
17.1.	Einführung	361
17.2.	Lernziele	361
17.3.	Übersicht über die Plugin-Architektur	361
17.4.	Installation des QGIS-Plugins	362
17.5.	Einrichtung der Abhängigkeiten	365
17.6.	Einrichtung des SAM-3-Zugangs	369
17.7.	GPU-Speicherverwaltung	370
17.8.	Prüfung auf Plugin-Updates	371
17.9.	Wichtigste Erkenntnisse	372
17.10.	Übungen	373
18.	<i>Baumsegmentierung in QGIS</i>	375
18.1.	Einführung	375
18.2.	Lernziele	375
18.3.	Überblick über das Baumsegmentierungs-Panel	376
18.4.	Laden vortrainierter Modelle	376
18.5.	Vorhersagen ausführen	378
18.6.	Ergebnisse exportieren	381
18.7.	Praktische Überlegungen	382
18.8.	Wichtigste Erkenntnisse	383
18.9.	Übungen	384
19.	<i>Wassersegmentierung in QGIS</i>	387
19.1.	Einleitung	387
19.2.	Lernziele	387
19.3.	Übersicht über das Wassersegmentierungs-Panel	388
19.4.	Konfiguration der Eingabedaten	388
19.5.	Durchführung der Wassersegmentierung	389
19.6.	Ausgabeoptionen	391
19.7.	Praktische Überlegungen	392
19.8.	Wichtige Erkenntnisse	393
19.9.	Übungen	394
20.	<i>Vision-Language-Modelle in QGIS</i>	395
20.1.	Einführung	395
20.2.	Lernziele	396
20.3.	Laden des Moondream-Modells	396
20.4.	Bildbeschriftung	398
20.5.	Visuelle Fragebeantwortung	400
20.6.	Objekterkennung	401
20.7.	Punktlokalisierung	403
20.8.	Tipps und Einschränkungen	404
20.9.	Wichtigste Erkenntnisse	405
20.10.	Übungen	405
21.	<i>Segment Anything in QGIS</i>	407

21.1.	Einführung	407
21.2.	Lernziele	407
21.3.	Beispieldaten	407
21.4.	Laden eines SAM-Modells	408
21.5.	Textgestützte Segmentierung	410
21.6.	Interaktive Segmentierung	411
21.7.	Stapelverarbeitung	413
21.8.	Ergebnisse exportieren	414
21.9.	Wichtigste Erkenntnisse	416
21.10.	Übungen	417
22.	<i>Semantische Segmentierung in QGIS</i>	419
22.1.	Einführung	419
22.2.	Lernziele	419
22.3.	Erstellung von Trainingsdaten	420
22.4.	Training eines Modells	423
22.5.	Inferenz durchführen	426
22.6.	Behebung häufiger Probleme	427
22.7.	Wichtige Erkenntnisse	428
22.8.	Übungen	428
23.	<i>Instanzsegmentierung in QGIS</i>	431
23.1.	Einführung	431
23.2.	Lernziele	431
23.3.	Mask R-CNN-Architektur	432
23.4.	Trainingsdaten erstellen	432
23.5.	Mask R-CNN trainieren	434
23.6.	Inferenz durchführen	436
23.7.	Wichtige Erkenntnisse	438
23.8.	Übungen	438
Literaturverzeichnis		441

Vorwort

Einführung

Wir befinden uns inmitten einer Revolution in der Erdbeobachtung. Jeden Tag erfassen Satelliten ein enormes Datenvolumen, darunter Multispektralbilder, Radar, LiDAR und mehr. Obwohl diese Informationen den Schlüssel zum Verständnis von Klimaanfälligkeit und Urbanisierung enthalten, ist die Lücke zwischen Datenerfassung und praktischer Anwendung eine große Hürde geblieben.

Hier kommt GeoAI ins Spiel.

Als Schnittstelle von Geoinformatik und KI schließt GeoAI diese Lücke in einem unglaublichen Tempo. Deep-Learning-Modelle können nun jedes Gebäude im großen Maßstab identifizieren oder einzelne Fahrzeuge über weite Gebiete hinweg lokalisieren. Neue Foundation-Modelle ermöglichen es diesen Werkzeugen, in unbekannten Umgebungen mit minimalem menschlichen Eingriff zu arbeiten. Darüber hinaus ermöglichen Vision-Language-Modelle und autonome Agenten den Analysten, Bildmaterial abzufragen und Workflows mit einfachen Befehlen zu verwalten. Die manuellen Digitalisierungsaufgaben, die einst Monate dauerten, werden heute in wenigen Minuten erledigt.

Dieses Buch, „*GeoAI with Python: A Practical Guide to Open-Source Geospatial AI*“, ist Ihr praktischer Leitfaden zu dieser Revolution. Es basiert auf einer einfachen Prämisse: Der beste Weg, GeoAI zu lernen, ist es zu tun. Anstatt sich mit abstrakter Theorie aufzuhalten, gibt Ihnen jedes Kapitel echte Werkzeuge an die Hand – von der Vorbereitung von Trainingsdaten und dem Ausführen von Deep-Learning-Modellen bis hin zur interaktiven Visualisierung und Interpretation Ihrer Ergebnisse. Der Code ist echt, die Datensätze sind echt, und die Problemstellungen spiegeln wider, was Praktiker täglich erleben.

Das Python-Ökosystem für GeoAI ist dramatisch gereift. Bibliotheken wie PyTorch, torchgeo und segment-geospatial bringen modernste Deep-Learning-Methoden zu Geoinformatik-Praktikern, während Pakete wie [leafmap](#) und [geoai](#) Hochlevel-Schnittstellen bieten, die komplexe Workflows zugänglich machen. Cloud-Computing-Plattformen und GPU-Beschleunigung haben die Hürde für das Training und den Einsatz von Modellen im großen Maßstab gesenkt. Zusammen bedeuten diese Fortschritte, dass ein Forscher mit einem Laptop heute das erreichen kann, wofür früher ein Team von Spezialisten und ein Raum voller Server nötig waren.

Unsere Reise beginnt mit den Grundlagen: Verständnis von Geodatenformaten, Einrichtung einer Deep-Learning-Umgebung und Beherrschung interaktiver Visualisierung. Anschließend behandeln wir die wesentliche Arbeit der Beschaffung von Satellitenbildern und der Vorbereitung von Trainingsdaten. Von dort aus gehen wir durch die zentralen KI-Aufgaben, die die moderne Fernerkundung definieren: Bilderkennung, Objekterkennung, semantische Segmentierung, Instanzsegmentierung, Bildübersetzung, Veränderungserkennung und pixelbasierte Regression. Dann erkunden wir die Grenzen der Foundation-Modelle, einschließlich des Segment Anything Model (SAM), Vision-Language-Modellen und Satelliteneinbettungen. Schließlich bringt ein eigener Abschnitt über QGIS-Plugins diese KI-Fähigkeiten in eine vertraute Desktop-GIS-Umgebung und ermöglicht anspruchsvolle Workflows ohne Programmierung.

Egal, ob Sie ein GIS-Fachmann sind, der KI in seine Workflows integrieren möchte, ein Datenwissenschaftler, der sich für Geoinformatik-Anwendungen interessiert, ein Forscher, der die Grenzen der Erdbeobachtung vorantreibt, oder ein Student, der eine Karriere an der Schnittstelle von Geografie und maschinellem Lernen beginnt – dieses Buch wird Sie mit dem Wissen und den Fähigkeiten ausstatten, Satellitenbilder in Erkenntnisse zu verwandeln.

Die Zukunft der Geodatenanalyse ist intelligent, automatisiert und zugänglich. Lassen Sie uns sie gemeinsam aufbauen.

Für wen dieses Buch ist

Dieses Buch richtet sich an alle, die künstliche Intelligenz auf Geodaten anwenden möchten. Wenn Sie jemals auf ein Satellitenbild gestarrt und sich gefragt haben, wie man Gebäude, Straßen oder Landbedeckung automatisch extrahieren kann, oder wenn Sie Deep-Learning-Modelle trainiert, aber Schwierigkeiten hatten, sie auf geografische Daten mit Projektionen, Koordinaten und riesigen Dateigrößen anzuwenden, dann ist dieses Buch für Sie.

Sie profitieren am meisten, wenn Sie

Ein GIS-Fachmann sind, der über manuelle Digitalisierung und visuelle Interpretation hinausgehen möchte. Sie sind versiert im Umgang mit QGIS oder ArcGIS und verstehen Konzepte der Raumanalyse, aber Sie möchten Deep Learning nutzen, um Merkmalsextraktion zu automatisieren, Landbedeckung im großen Maßstab zu klassifizieren oder Veränderungen in Zeitreihenbildern zu erkennen.

Ein Fernerkundungswissenschaftler oder Forscher, der mit Satelliten- oder Luftbildern arbeitet. Sie verstehen Spektralbänder, räumliche Auflösung und Bildvorverarbeitung, aber Sie brauchen eine praktische Brücke zu modernen KI-Techniken – vom Training von Segmentierungsmodellen bis zur Anwendung von Foundation-Modellen wie SAM auf Ihre Untersuchungsgebiete.

Ein Datenwissenschaftler oder Machine-Learning-Ingenieur mit Erfahrung im Deep Learning, der seine Fähigkeiten auf Geoinformatik-Probleme anwenden möchte. Sie sind vertraut mit PyTorch oder TensorFlow und verstehen CNNs und Transformer, aber Sie brauchen Anleitung zu den besonderen Herausforderungen geografischer Daten: Koordinatenreferenzsysteme, Kachelung großer Raster, Umgang mit Mehrkanalbildern und Georeferenzierung von Modellergebnissen.

Ein Doktorand oder Nachwuchsforscher in Geografie, Umweltwissenschaften, Stadtplanung, Ökologie oder einem verwandten Fachgebiet. Ihre Forschung umfasst räumliche Daten, und Sie möchten modernste KI-Methoden in Ihre Abschlussarbeit oder Publikationen integrieren und gleichzeitig Fähigkeiten aufbauen, die zunehmend gefragt sind.

Ein Softwareentwickler, der Geoinformatik-Anwendungen mit intelligenter Analyse erstellt. Sie müssen KI-gestützte Merkmalsextraktion, Klassifikation oder Veränderungserkennung in Webanwendungen, APIs oder automatisierte Pipelines integrieren und möchten den gesamten Workflow vom Modelltraining bis zum Deployment verstehen.

Grundlegende Voraussetzungen

Sie sollten vertraut sein mit:

- **Python-Programmierung:** Variablen, Funktionen, Klassen und das Importieren von Bibliotheken (fortgeschrittene Python-Kenntnisse sind nicht erforderlich)
- **Grundlegende Datenanalyse:** Arbeiten mit tabellarischen Daten, Filtern, Aggregieren und Visualisieren
- **Grundlegende Geoinformatik-Konzepte:** Verständnis, dass Daten einen Standort haben, was Raster- und Vektordaten sind, und grundlegende Vertrautheit mit Koordinatensystemen
- **Grundlagen der Kommandozeile:** Navigation in Verzeichnissen, Ausführen von Skripten und Installieren von Paketen

Hilfreicher Hintergrund (aber nicht erforderlich)

- Erfahrung mit Deep-Learning-Frameworks (PyTorch, TensorFlow)

- Vertrautheit mit Fernerkundungskonzepten (Spektralbänder, räumliche Auflösung, Bildklassifikation)
- Vorerfahrung mit geoinformatischen Python-Bibliotheken (rasterio, geopandas, leafmap)
- Verständnis der Grundlagen des maschinellen Lernens (Training vs. Inferenz, Überanpassung, Evaluationsmetriken)

Wenn Sie neu in der geoinformatischen Python-Programmierung sind

Wenn Sie neu in der geoinformatischen Python-Programmierung sind, bietet das folgende Buch eine hervorragende Einführung in grundlegende GIS-Konzepte und Python-Programmierung:

Wu, Q. (2025). *Introduction to GIS Programming: A Practical Python Guide to Open Source Geospatial Tools*. Independently published. PDF edition ISBN 9798993859712; Print edition ISBN 979-8286979455. Verfügbar unter gispro.gishub.org.

Für diejenigen, die sich für räumliches Datenmanagement und SQL-basierte Geodatenanalyse interessieren, bietet das Begleitbuch einen umfassenden Leitfaden:

Wu, Q. (2025). *Spatial Data Management with DuckDB*. Independently published. PDF edition ISBN 979-8993859705; Print edition ISBN 979-8274710572. Verfügbar unter duckdb.gishub.org.

Was dieses Buch behandelt

Dieses Buch bietet eine strukturierte Reise von den Grundlagen der Geoinformatik bis hin zu produktionsreifen GeoAI-Anwendungen und vermittelt Ihnen bei jedem Schritt praktische Fähigkeiten durch praxisnahe Beispiele. Jedes Kapitel baut auf dem vorherigen auf und erweitert schrittweise Ihre Fähigkeit, KI auf reale Geoinformatik-Probleme anzuwenden.

Teil I: Grundlagen (*Kapitel 1–3*)

Vermittlung des grundlegenden Wissens und der Werkzeuge, die allen nachfolgenden Inhalten zugrunde liegen:

- **Kapitel 1: Einführung in GeoAI** bietet einen umfassenden Überblick über die GeoAI-Landschaft: was es ist, warum es wichtig ist und wie Deep Learning die Geodatenanalyse transformiert hat. Sie werden die wichtigsten KI-Aufgaben in der Fernerkundung erkunden, das Python-Ökosystem für GeoAI überblicken und verstehen, wohin Foundation-Modelle und KI-Agenten das Feld führen.
- **Kapitel 2: Einrichtung der Umgebung** führt Sie durch die Konfiguration einer vollständigen GeoAI-Entwicklungsumgebung. Von der Installation von Python und der Paketverwaltung mit conda bis zur Einrichtung der GPU-Beschleunigung mit CUDA und PyTorch stellt dieses Kapitel sicher, dass Sie eine solide, reproduzierbare Grundlage für alle folgenden praktischen Arbeiten haben.
- **Kapitel 3: Geodatenformate** behandelt die Datenformate, denen Sie im gesamten Buch begegnen werden, einschließlich Rasterformate wie GeoTIFF und Cloud Optimized GeoTIFF (COG), Vektorformate wie GeoJSON und GeoParquet sowie spezialisierte Formate für Deep Learning wie COCO- und Pascal-VOC-Annotationen. Sie lernen, Formate mit Python zu lesen, zu schreiben und zwischen ihnen zu konvertieren.

Am Ende von Teil I verfügen Sie über eine vollständig konfigurierte Deep-Learning-Umgebung und ein solides Verständnis der Datenformate und Konzepte, die für die Arbeit mit GeoAI erforderlich sind.

Teil II: Datenerfassung und -vorbereitung (*Kapitel 4–6*)

Beherrschen Sie die kritische, aber oft unterschätzte Arbeit der Beschaffung und Vorbereitung von Geodaten für KI:

- **Kapitel 4: Herunterladen von Fernerkundungsdaten** lehrt Sie, wie Sie programmatisch auf Satellitenbilder, Höhendaten und andere Geodatenätze aus Quellen wie Google Earth Engine, Microsoft Planetary Computer und verschiedenen offenen Datenportalen zugreifen können. Sie lernen, Daten für Ihre Untersuchungsgebiete effizient zu suchen, zu filtern und herunterzuladen.
- **Kapitel 5: Interaktive Kartierung und Visualisierung** stellt leistungsstarke Visualisierungswerkzeuge für die interaktive Erkundung von Geodaten vor. Mit leafmap und anderen Bibliotheken erstellen Sie interaktive Karten, überlagern Satellitenbilder, visualisieren Modellvorhersagen und erstellen überzeugende visuelle Darstellungen. Diese Fähigkeiten sind sowohl für die explorative Analyse als auch für die Kommunikation von Ergebnissen unerlässlich.
- **Kapitel 6: Erstellen von Trainingsdaten** behandelt einen der wichtigsten Schritte in jeder KI-Pipeline: die Vorbereitung hochwertiger Trainingsdatensätze. Sie lernen, gelabelte Datensätze für Segmentierungs-, Erkennungs- und Klassifikationsaufgaben zu erstellen, einschließlich Annotationstrategien und dem Kacheln großer Satellitenbilder in Trainings-Chips.

Am Ende von Teil II können Sie Satellitenbilder aus mehreren Quellen beschaffen, sie interaktiv visualisieren und gut strukturierte Trainingsdatensätze vorbereiten, die für Deep-Learning-Modelle bereit sind.

Teil III: Zentrale KI-Aufgaben (*Kapitel 7–13*)

Tauchen Sie ein in die grundlegenden KI-Aufgaben, die die moderne Geodatenanalyse definieren:

- **Kapitel 7: Bilderkennung** führt in Deep Learning zur Klassifikation ganzer Bilder ein. Sie lernen, Bildklassifikationsmodelle zu erstellen, die Szenentypen, Landnutzungskategorien und geografische Merkmale aus Satelliten- und Luftbildern erkennen.
- **Kapitel 8: Objekterkennung** lehrt Sie, einzelne Objekte in Geodatenbildern zu lokalisieren und zu klassifizieren. Von der Erkennung von Gebäuden und Fahrzeugen bis zur Identifikation von Bäumen und Solarpanelen arbeiten Sie mit Architekturen wie Faster R-CNN und YOLO, die für die Fernerkundung angepasst wurden.
- **Kapitel 9: Semantische Segmentierung** behandelt die pixelweise Klassifikation, bei der jedem Pixel in einem Bild ein Klassenlabel zugewiesen wird. Sie trainieren Modelle zur Abgrenzung von Merkmalen wie Gewässern, Vegetation und bebauten Flächen aus Satellitenbildern und lernen dabei Architekturen wie U-Net und DeepLabV3+.
- **Kapitel 10: Instanzsegmentierung** kombiniert Erkennung und Segmentierung und erzeugt präzise Grenzen für jedes einzelne Objekt. Sie lernen, überlappende Merkmale zu unterscheiden, wie z.B. einzelne Gebäudegrundrisse in dicht bebauten städtischen Gebieten, unter Verwendung von Modellen wie Mask R-CNN.
- **Kapitel 11: Bildübersetzung** wendet KI an, um Bilder zwischen Domänen zu transformieren, einschließlich Superauflösung, Sensorübersetzung und Wolkenentfernung. Sie trainieren Modelle mit Multispektral-Satellitenbildern und lernen, höher aufgelöste Ausgaben aus niedriger aufgelösten Eingaben zu generieren.
- **Kapitel 12: Veränderungserkennung** befasst sich mit der Herausforderung, zu identifizieren, was sich zwischen Bildern verändert hat, die zu verschiedenen Zeitpunkten aufgenommen wurden. Von Stadterweiterung und Entwaldung bis zur Schadensbewertung nach Katastrophen lernen Sie sowohl traditionelle als auch Deep-Learning-Ansätze zur zeitlichen Analyse.

- **Kapitel 13: Pixelbasierte Regression** geht über die Klassifikation hinaus und sagt kontinuierliche Werte für jedes Pixel vorher, wie Kronenhöhe, Biomasse, Bodenfeuchte oder Bevölkerungsdichte. Sie trainieren Regressionsmodelle auf Satellitenbildern und lernen Evaluationsstrategien, die speziell für kontinuierliche Vorhersagen gelten.

Am Ende von Teil III haben Sie praktische Erfahrung mit allen wichtigen GeoAI-Aufgaben, verstehen, wann welcher Ansatz anzuwenden ist, und können Modelle für Ihre eigenen Geoinformatik-Anwendungen trainieren und evaluieren.

Teil IV: Foundation-Modelle und Satelliteneinbettungen (*Kapitel 14–16*)

Erkunden Sie die Spitze der GeoAI, wo vortrainierte Foundation-Modelle neu definieren, was möglich ist:

- **Kapitel 14: Segment Anything für Geodaten** stellt das Segment Anything Model (SAM) und seine Anwendung auf Geodatenbilder vor. Mit dem [segment-geospatial](#)-Paket lernen Sie, Satellitenbilder mit minimalen Eingabeaufforderungen zu segmentieren und Gebäude, landwirtschaftliche Flächen, Gewässer und mehr ohne aufgabenspezifisches Training zu extrahieren.
- **Kapitel 15: Vision-Language-Modelle** erkundet Modelle, die visuelles und textuelles Verständnis verbinden. Sie lernen, Satellitenbilder mit natürlicher Sprache abzufragen, Beschriftungen für Geodaten szenen zu generieren und multimodale Modelle für Aufgaben wie visuelle Fragebeantwortung bei Fernerkundungsdaten einzusetzen.
- **Kapitel 16: Satelliteneinbettungen** erkundet das rasant wachsende Ökosystem vorberechneter Satelliteneinbettungsdatensätze aus Foundation-Modellen. Sie lernen, Einbettungen aus neun Datensätzen wie Clay, TESSERA und Googles AlphaEarth zu durchsuchen, zu laden und zu visualisieren, Ähnlichkeitssuche und Clustering durchzuführen und leichtgewichtige Klassifikatoren auf Einbettungsvektoren zu trainieren.

Am Ende von Teil IV verstehen Sie, wie Foundation-Modelle GeoAI umgestalten, und Sie sind in der Lage, diese fortgeschrittenen Techniken auf Ihre eigene Forschung und Projekte anzuwenden.

Teil V: QGIS-Plugins (*Kapitel 17–23*)

Bringen Sie GeoAI-Fähigkeiten in die vertraute QGIS-Desktop-GIS-Umgebung über das GeoAI-Plugin:

- **Kapitel 17: Einrichtung des GeoAI-QGIS-Plugins** führt durch die Installation und Konfiguration des GeoAI-QGIS-Plugins. Sie verwenden den integrierten Abhängigkeitsinstallator oder richten eine Pixi-Umgebung mit PyTorch- und CUDA-Unterstützung ein, installieren das Plugin und lernen, den GPU-Speicher für die Ausführung großer KI-Modelle in QGIS zu verwalten.
- **Kapitel 18: Baumsegmentierung in QGIS** demonstriert das Baumsegmentierungspanel für die Erkennung von Baumkronen und ökologischen Objekten. Sie laden vortrainierte Modelle für Bäume, Vögel, Vieh, Nester und tote Bäume, konfigurieren Vorhersageparameter für einzelne Bilder und große Kacheln und exportieren Ergebnisse in Vektor-, Raster- und Trainingsdatenformate.
- **Kapitel 19: Wassersegmentierung in QGIS** behandelt das Wassersegmentierungspanel für die automatisierte Gewässererkennung mit OmniWaterMask. Sie konfigurieren die Bandreihenfolge für verschiedene Sensoren, optimieren Parameter für die kachelbasierte Inferenz, integrieren OpenStreetMap-Daten zur Verfeinerung und exportieren Wassermasken als Raster- und Vektordaten.
- **Kapitel 20: Vision-Language-Modelle in QGIS** demonstriert das Moondream-VLM-Panel für die natürlichsprachliche Interaktion mit Geodatenbildern. Sie generieren Bildbeschriftungen, stellen Fragen zu sichtbaren Merkmalen, erkennen Objekte mit Begrenzungsrahmen und lokalisieren Merkmale mit Punktmarkierungen.

- **Kapitel 21: Segment Anything in QGIS** demonstriert das Segment-Anything-Panel für interaktive und automatisierte Segmentierung. Sie laden SAM 3, segmentieren Objekte mit Texteingabeaufforderungen, führen interaktive Segmentierung mit Punkt- und Rahmen-Prompts durch, verarbeiten Stapel von Merkmalen und exportieren georeferenzierte Ergebnisse.
- **Kapitel 22: Semantische Segmentierung in QGIS** behandelt den vollständigen Workflow vom Training bis zur Inferenz für pixelweise Klassifikation ohne Programmierung. Sie erstellen Trainingsdatensätze, wählen aus Architekturen wie U-Net, DeepLabV3+ und SegFormer, trainieren benutzerdefinierte Modelle und führen Inferenz mit Vektorisierung und Glättung durch.
- **Kapitel 23: Instanzsegmentierung in QGIS** demonstriert instanzbasierte Objekterkennung mit Mask R-CNN. Sie erstellen Instanz-Trainingsdaten im PASCAL_VOC-Format, trainieren Modelle zur Erkennung und Abgrenzung einzelner Objekte und nutzen die objektweise Ausgabe für Zählung, Messung und räumliche Analyse.

Am Ende von Teil V können Sie anspruchsvolle GeoAI-Workflows direkt in QGIS ausführen und KI-gestützte Analysen für GIS-Praktiker zugänglich machen, ohne Python-Programmierkenntnisse zu benötigen.

Übergreifende Themen

- **Praktisches Arbeiten:** Jedes Konzept wird von ausführbaren Codebeispielen mit realen Geodaten begleitet.
- **Open-Source-Tools:** Alle im Buch verwendeten Programme sind frei und quelloffen, was Zugänglichkeit und Reproduzierbarkeit gewährleistet.
- **Skalierbare Workflows:** Techniken, die auf einem Laptop funktionieren und auf Cloud-Infrastruktur skalieren.
- **Praxisnahe Anwendungen:** Beispiele aus Umweltmonitoring, Stadtanalyse, Landwirtschaft und Katastrophenhilfe.
- **Reproduzierbarkeit:** Alle Code- und Datenbeispiele sind auf GitHub verfügbar, sodass Sie jedes Ergebnis im Buch reproduzieren können.

Das Beste aus diesem Buch herausholen

Um Ihre Lernerfahrung mit diesem Buch zu maximieren, beachten Sie die folgenden Empfehlungen:

Richten Sie Ihre Umgebung frühzeitig ein: Folgen Sie Kapitel 2 sorgfältig, um Ihre Python-Umgebung mit GPU-Unterstützung zu konfigurieren. Viele GeoAI-Aufgaben profitieren erheblich von GPU-Beschleunigung, und eine richtig konfigurierte Umgebung wird Ihnen im gesamten Buch erheblich Zeit sparen. Wenn Sie keine lokale GPU haben, bietet das Buch Anleitungen zur Nutzung cloudbasierter GPU-Plattformen wie Google Colab.

Arbeiten Sie mit dem Code mit: Dieses Buch ist interaktiv konzipiert. Lesen Sie den Code nicht nur – tippen Sie ihn ab, führen Sie ihn aus und beobachten Sie die Ergebnisse. Ändern Sie Parameter, probieren Sie verschiedene Datensätze aus und experimentieren Sie. Verständnis entsteht durch Praxis, und die praktischen Beispiele sind das Herzstück dieses Buches. Wenn etwas nicht wie erwartet funktioniert, widerstehen Sie dem Drang, weiterzuspringen. Debugging ist eine der wertvollsten Lernerfahrungen, und die Fehlerbehebungsfähigkeiten, die Sie entwickeln, werden Ihnen während Ihrer gesamten GeoAI-Karriere dienen.

Arbeiten Sie die Kapitel der Reihe nach durch: Obwohl erfahrene Praktiker zu bestimmten Themen springen können, bauen die Kapitel aufeinander auf. Konzepte, die in früheren Kapiteln eingeführt

werden (Umgebungseinrichtung, Datenformate, Visualisierungstechniken), werden im gesamten Buch verwendet. Wenn Sie vorspringen, schauen Sie bei Bedarf zurück.

Verwenden Sie Ihre eigenen Daten: Obwohl das Buch Datensätze für jedes Beispiel bereitstellt, geschieht das wirkliche Lernen, wenn Sie diese Techniken auf Daten anwenden, die Ihnen wichtig sind. Versuchen Sie, die Segmentierungsmodelle auf Bildmaterial Ihres Untersuchungsgebiets auszuführen, oder trainieren Sie ein Objekterkennungsmodell für Merkmale, die für Ihre Forschung relevant sind.

Akzeptieren Sie Fehler: Das Training von Deep-Learning-Modellen erfordert Iteration. Modelle werden nicht immer beim ersten Versuch konvergieren, Vorhersagen werden nicht immer perfekt sein, und GPUs werden gelegentlich keinen Speicher mehr haben. Dies sind Lernmöglichkeiten. Das Buch behandelt häufige Fallstricke und Debugging-Strategien, aber das Gespür für die Diagnose von Problemen entwickelt sich durch Erfahrung.

Erstellen Sie ein Portfolio-Projekt: Identifizieren Sie im Laufe des Buches ein Geoinformatik-Problem, das Sie interessiert, und wenden Sie die Techniken an, die Sie lernen. Ein vollständiges Projekt – von der Datenbeschaffung über das Modelltraining bis zur Visualisierung – demonstriert Ihre Fähigkeiten weitaus effektiver als jede Zertifizierung.

Bleiben Sie aktuell: GeoAI ist ein sich schnell entwickelndes Feld. Neue Foundation-Modelle, Architekturen und Werkzeuge erscheinen regelmäßig. Das GitHub-Repository des Buches wird periodisch mit neuen Inhalten und Beispielen aktualisiert, um Ihre Lernreise fortzusetzen.

In diesem Buch verwendete Konventionen

Dieses Buch verwendet verschiedene Konventionen, um Ihnen bei der Navigation durch den Inhalt und dem Verständnis der Codebeispiele zu helfen:

Code-Formatierung: Jeder Python-Code erscheint in einer Monospace-Schrift innerhalb von Codeblöcken. Wenn Code im regulären Text erscheint, ist er so formatiert wie hier. Datei- und Verzeichnisnamen, Paketnamen und Funktionsnamen sind ebenfalls in Monospace-Schrift formatiert.

Codebeispiele: Die meisten Codebeispiele sind vollständig und ausführbar. Sie enthalten Kommentare, die die wichtigsten Konzepte und Techniken erklären. Hier ist ein typisches Beispiel mit der `leafmap`-Bibliothek:

```
import leafmap

m = leafmap.Map(center=[40, -100], zoom=4)
m.add_basemap("Esri.WorldImagery")
m
```

Paket-Importe: Am Anfang jedes Kapitels werden alle erforderlichen Importe aufgelistet. Wichtige Pakete, die Ihnen im gesamten Buch begegnen werden, sind:

- `geoai` - Hochlevel-GeoAI-Workflows (opengeoai.org)
- `leafmap` - Interaktive geoinformatische Visualisierung (leafmap.org)
- `samgeo` - SAM für Geodaten (samgeo.github.org)
- `torch` / `torchvision` / `torchgeo` - Deep Learning mit PyTorch
- `rasterio` / `geopandas` - Lesen und Schreiben von Geodaten

Abbildungen und Karten: Interaktive Karten werden direkt in Jupyter-Notebooks gerendert. In der gedruckten Ausgabe werden statische Momentaufnahmen mit Bildunterschriften gezeigt, die die interaktiven Funktionen beschreiben, die in der Online-Version verfügbar sind.

Herunterladen der Codebeispiele

Alle Codebeispiele und Begleitmaterialien für dieses Buch sind frei auf GitHub verfügbar: <https://github.com/giswqs/GeoAI-Book>

Alle Beispieldatensätze sind bei Source Cooperative verfügbar: <https://source.coop/opengeos/geoai>

Um die Materialien herunterzuladen, können Sie eine der folgenden Methoden verwenden:

- **Repository klonen** (empfohlen, wenn Git installiert ist):

```
$ git clone https://github.com/giswqs/GeoAI-Book.git
```

- **Als ZIP herunterladen** (wenn Sie Git nicht verwenden möchten):
 - Besuchen Sie die GitHub-Repository-Seite
 - Klicken Sie auf den grünen **Code**-Button
 - Wählen Sie **Download ZIP**
 - Entpacken Sie die Dateien an Ihrem bevorzugten Speicherort
- **Einzelne Dateien durchsuchen** online über die GitHub-Oberfläche oder die Buch-Website unter book.opengeoai.org

Das Repository wird regelmäßig mit Korrekturen, Verbesserungen und zusätzlichen Beispielen aktualisiert. Schauen Sie regelmäßig vorbei, oder **markieren** und **beobachten** Sie das Repository auf GitHub, um über Änderungen informiert zu werden.

Wenn Sie Fehler im Code finden oder Verbesserungsvorschläge haben, öffnen Sie bitte ein Issue oder reichen Sie einen Pull Request auf GitHub ein. Beiträge aus der Community helfen, diese Ressource für alle zu verbessern.

Video-Tutorials

Ergänzend zum geschriebenen Inhalt wird dieses Buch durch Video-Tutorials unterstützt, die Sie durch wichtige Konzepte führen und zusätzliche Demonstrationen bieten:

<https://youtube.com/@giswqs>

Die Videos sind als Ergänzung zum geschriebenen Material gedacht, nicht als Ersatz. Sie sind besonders hilfreich für:

- Visuelle Lerner, die davon profitieren, Code in Echtzeit geschrieben und ausgeführt zu sehen
- Das Verständnis komplexer Workflows wie Modelltraining und Inferenz durch schrittweise Anleitungen
- Das Erlernen des Umgangs mit Geoinformatik-Problemen und das Debuggen häufiger Fehler
- Tipps und Best Practices für die Arbeit mit großen Satellitendatensätzen

Weitere Video-Tutorials werden hinzugefügt, wenn sich das Buch weiterentwickelt. Abonnieren Sie den Kanal, um über neue Inhalte informiert zu werden.

Community und Feedback

Ich begrüße Feedback, Fragen und Vorschläge von Lesern. Ihre Beiträge helfen, das Buch zu verbessern und es für die GeoAI-Community nützlicher zu machen.

Für buchbezogene Fragen und Diskussionen:

- GitHub Issues: <https://github.com/giswqs/GeoAI-Book/issues>
- GitHub Discussions: <https://github.com/giswqs/GeoAI-Book/discussions>

Für paketspezifische Fragen:

- geoai: <https://github.com/opengeos/geoai/issues>
- leafmap: <https://github.com/opengeos/leafmap/issues>
- segment-geospatial: <https://github.com/opengeos/segment-geospatial/issues>

Besonders hilfreiche Arten von Feedback:

- Fehler oder unklare Erklärungen im Text oder Code
- Vorschläge für zusätzliche Beispiele, Datensätze oder Anwendungsfälle
- Berichte über Kompatibilitätsprobleme mit verschiedenen Betriebssystemen, GPU-Konfigurationen oder Bibliotheksversionen
- Ideen für neue Themen oder aufkommende GeoAI-Techniken
- Erfolgsgeschichten, wie Sie die Techniken aus dem Buch auf Ihre eigene Arbeit angewendet haben

Über den Autor

Dr. Qiusheng Wu ist Associate Professor und Direktor der Graduiertenstudien am Department of Geography & Sustainability der University of Tennessee, Knoxville. Er ist außerdem als Amazon Scholar tätig. Dr. Wus Forschung konzentriert sich auf die Weiterentwicklung von Open-Source-Geodatenanalyse durch Cloud Computing und GeoAI, mit Schwerpunkt auf der Nutzung von Big Geospatial Data und künstlicher Intelligenz zur Untersuchung von Umweltveränderungen, insbesondere der Dynamik von Oberflächenwasser und Feuchtgebietsinundation.

Er ist der Schöpfer und Betreuer mehrerer weit verbreiteter Open-Source-Python-Pakete, darunter [geemap](https://geemap.org)¹ für interaktive Google-Earth-Engine-Visualisierung, [leafmap](https://leafmap.org)² für vielseitige geoinformatische Kartierung, [segment-geospatial](https://samgeo.gishub.org)³ für die Anwendung des Segment Anything Model auf Geodaten und [geoai](https://opengeoai.org)⁴ für Hochlevel-GeoAI-Workflows. Seine Open-Source-Projekte, verfügbar über die [Open Geospatial Solutions](https://github.com/opengeos)⁵-Organisation auf GitHub, werden weltweit von Forschern, Lehrenden und Praktikern eingesetzt.

Dr. Wus Arbeit verbindet Fernerkundung, Erdbeobachtung und künstliche Intelligenz, um großskalige Geodaten zugänglicher, reproduzierbarer und intelligenter zu machen. Er ist leidenschaftlich für offene Wissenschaft und glaubt, dass die besten Werkzeuge zum Verständnis unseres Planeten für jeden frei verfügbar sein sollten.

¹<https://geemap.org>

²<https://leafmap.org>

³<https://samgeo.gishub.org>

⁴<https://opengeoai.org>

⁵<https://github.com/opengeos>

Lizenzierung und Urheberrecht

Dieses Buch folgt den Prinzipien offener Wissenschaft und offener Bildung. Um Transparenz, Lernen und Wiederverwendung zu unterstützen, werden die **Codebeispiele** in diesem Buch unter einer [Creative Commons Attribution 4.0 International \(CC BY 4.0\)](#)-Lizenz veröffentlicht. Das bedeutet, Sie dürfen den Code frei kopieren, ändern und verbreiten, auch für kommerzielle Zwecke, solange eine angemessene Quellenangabe erfolgt.

Bitte zitieren Sie bei der Nutzung des Codes das Buch oder verlinken Sie auf das GitHub-Repository:

Wu, Q. (2026). *GeoAI with Python: A Practical Guide to Open-Source Geospatial AI*. Independently published. PDF edition ISBN 979-8993859729; Print edition ISBN 979-8253507414. Verfügbar unter book.opengeoai.org.

Obwohl der Code frei verfügbar ist, sind der **Text, die Abbildungen und Bilder** in diesem Buch **urheberrechtlich geschützt** © 2026 durch den Autor und dürfen ohne ausdrückliche Genehmigung nicht reproduziert, weiterverbreitet oder verändert werden. Dies umfasst alle geschriebenen Inhalte, benutzerdefinierten Diagramme und eingebetteten Visualisierungen, sofern nicht anders angegeben.

Wenn Sie Nicht-Code-Material aus dem Buch wiederverwenden oder anpassen möchten (zum Beispiel für Lehre, Präsentationen oder Publikationen), kontaktieren Sie bitte den Autor, um eine Genehmigung zu beantragen.

Dieser duale Lizenzansatz hilft, den offenen Zugang zu Lernmaterialien mit dem Schutz originaler kreativer Arbeit in Einklang zu bringen. Vielen Dank, dass Sie diese Bedingungen respektieren und die Open-Source-Geoinformatik-Community unterstützen.

Ein praxisorientierter Leitfaden für Open-Source-Geodaten-KI

Künstliche Intelligenz verändert die Art und Weise, wie wir unseren Planeten analysieren und verstehen. **GeoAI mit Python** ist der maßgebliche praxisorientierte Leitfaden zur Anwendung von KI-Techniken auf geospatiale Daten, von Satellitenbildern und Luftaufnahmen bis hin zu Vektordatensätzen und Erdbeobachtungsarchiven.

Jedes Kapitel enthält ein begleitendes Jupyter-Notebook mit realen Datensätzen, sodass Sie Schritt für Schritt mitarbeiten und alle Ergebnisse reproduzieren können. Ob Sie GIS-Experte sind und KI erkunden oder als Data Scientist erstmals mit räumlichen Daten arbeiten, dieses Buch verbindet beide Welten.

Qiusheng Wu ist außerordentlicher Professor an der University of Tennessee und Amazon Scholar. Er ist der Entwickler mehrerer weit verbreiteter Open-Source-Pythonpakete für Geodaten, darunter geemap, leafmap, segment-geospatial und geoai. Seine Projekte sind unter <https://github.com/opengeos> verfügbar.

Was Sie lernen werden

- Fernerkundungsdaten aus NAIP, Sentinel-2 und STAC-Katalogen herunterladen und aufbereiten
- Deep-Learning-Modelle für Segmentierung, Objekterkennung und Landbedeckungsklassifikation trainieren
- Foundation-Modelle wie SAM 3, Moondream und Satelliten-Embeddings auf reale Anwendungsfälle anwenden
- Interaktive Karten und Visualisierungen für geospatiale Analysen erstellen
- QGIS-Plugins für Segmentierung, Training, Inferenz und Vision-Language-Modelle nutzen



Scannen Sie den QR-Code, um die Buchwebsite zu besuchen und Codebeispiele von <https://book.opengeoai.org> herunterzuladen.

