

Was steckt in deinen Dateien?

Die Ebenen- Prüfliste

Das Repo watermarks-remover hat in rund 24 Tagen über 20.000 GitHub-Sterne gesammelt — und wirft dabei technisch sehr verschiedene Dinge in einen Topf. Diese Prüfliste trennt die drei Ebenen sauber: Du siehst für deine eigenen Dateien, welche unsichtbaren Zeichen, Metadaten und Provenance-Angaben wirklich drinstecken und wie du jede Ebene einzeln checkst.

3 EBENEN GETRENNT

VORHER/NACHHER-CHECK

LOKAL, KEINE CLOUD

NUR EIGENE INHALTE

Worum es geht: „AI-Wasserzeichen“ ist kein einzelnes Ding. In einer Datei können **drei technisch verschiedene Ebenen** stecken, die verschieden geprüft und verschieden behandelt werden. Wer sie vermischt, prüft am Ende gar nichts. Diese Liste geht die Ebenen einzeln durch — an deinen eigenen Dateien.

1 Die drei Ebenen im Überblick

EBENE	WAS ES IST	WIE ES SICH VERHÄLT
Unicode-Zeichen	Unsichtbare Zeichen im Text selbst, etwa Zero-Width-Spaces oder besondere Leerzeichen	Direkt auffindbar und direkt löschar
Metadaten & C2PA	Angaben in der Datei-Hülle: Erzeuger-Tool, Zeitstempel, Provenance-Kette (C2PA) in Bild, Video, Audio, PDF	Eigener Prüf- und Bearbeitungsschritt, unabhängig vom Inhalt
Statistische Muster	Verteilungen in Wortwahl und Satzbau, die ein Anbieter beim Erzeugen einprägt	Nur durch Umschreiben veränderbar — zuverlässig weder nachweisbar noch entfernbar

2 Ebene 1 prüfen: unsichtbare Unicode-Zeichen

Diese Zeichen sieht kein Auge, aber jeder Parser. Die üblichen Verdächtigen: Zero-Width-Space (U+200B), Zero-Width-Joiner und -Non-Joiner (U+200D, U+200C), Soft Hyphen (U+00AD) und Sonder-Leerzeichen wie U+00A0 oder U+2009.

So machst du sie im Terminal sichtbar:

```
# zeigt jedes Nicht-Standard-Zeichen mit Position und Codepoint
python3 -c "
import sys
for i, ch in enumerate(open(sys.argv[1], encoding='utf-8').read()):
    if ord(ch) > 126 and not ch.isalpha():
        print(i, hex(ord(ch)), repr(ch))
" deine-datei.txt
```

Kommt nichts zurück, ist die Ebene sauber. Treffer wie **0x200b** sind unsichtbare Zeichen — die kannst du gezielt ersetzen. Umlaute und normale Satzzeichen listet der Befehl zwar mit, die erkennst du aber sofort am **repr**.

Vorher/Nachher-Check: denselben Befehl vor und nach jeder Bearbeitung laufen lassen. Sauber heißt: null unsichtbare Treffer im Nachher-Lauf.

3 Ebene 2 prüfen: Metadaten und C2PA

Bei Bildern, Videos, Audio und PDFs steckt die Information nicht im Inhalt, sondern in der Datei-Hülle. Zwei Werkzeuge decken das ab:

```
# klassische Metadaten (EXIF, XMP): Erzeuger-Tool, Software, Zeitstempel
exiftool deine-datei.jpg

# C2PA-Provenance-Kette (offizielles CLI der C2PA-Initiative)
c2patool deine-datei.jpg
```

Interessant sind Felder wie **Software**, **Creator Tool** und jede C2PA-Manifest-Kette: Dort steht, welches Tool die Datei erzeugt oder bearbeitet hat. **exiftool** kommt über Homebrew, **c2patool** über die C2PA-GitHub-Releases.

Vorher/Nachher-Check: beide Ausgaben vor und nach der Bearbeitung vergleichen. Wichtig zu wissen: Ein leeres C2PA-Manifest beweist nicht, dass die Datei nie eines hatte — nur, dass jetzt keines mehr da ist.

4 Ebene 3: statistische Muster — die ehrliche Einordnung

Für statistische Text-Wasserzeichen gibt es keinen Selbst-Check, der Gewissheit liefert. Es ist nicht belegt, dass jeder Anbieter seine Ausgaben markiert, und ebenso wenig, dass ein Tool jede Markierung zuverlässig erkennt oder entfernt. Umschreiben verändert die Muster — mehr lässt sich seriös nicht zusagen.

Merksatz für diese Ebene: **„Wasserzeichen verändert“ und „von jeder Erkennung unauffindbar“ sind zwei verschiedene Aussagen.** Wer dir Zweiteres verspricht, verspricht etwas Unbelegbares.

5 Das Repo dahinter

guillaumemeyer/watermarks-remover packt die Verfahren in einen lokalen Python-Service mit Agent-Skill — deine Dateien bleiben auf deinem Rechner. Version 0.7.0 erschien am 3. September und adressiert Text-, Bild-, Dokument-, Audio- und Videoformate, C2PA eingeschlossen. Die 20.535 Sterne (Stand 4. September) entstanden in rund 24 Tagen.

Zwei Dinge belegen die Sterne nicht: die Wirksamkeit auf Ebene 3 und die Rechtmäßigkeit jedes Einsatzes. Genau deshalb gehört zu jedem Lauf der Vorher/Nachher-Check aus den Abschnitten 2 und 3.

RECHT & ETHIK Nutze das Tool ausschließlich für Dateien, deren Rechte bei dir liegen. Das

Entfernen von Provenance-Angaben aus fremden Inhalten kann rechtliche Folgen haben, und Kennzeichnungspflichten für KI-Inhalte (etwa auf Plattformen oder nach EU-Regeln) bestehen unabhängig davon, was technisch möglich ist. Diese Prüfliste ist eine technische Übersicht und keine Rechtsberatung.

6 Quellen

- **Repository:** github.com/guillaumemeyer/watermarks-remover
- **Release 0.7.0 (03.09.2026):** github.com/guillaumemeyer/watermarks-remover/releases/tag/v0.7.0
- **C2PA-CLI:** github.com/contentauth/c2patool
- **ExifTool:** exiftool.org

■ // AUF EINEN BLICK

Die Kurzfassung

Vier Punkte für jede eigene Datei.

- ✓ **Drei Ebenen, drei Checks** — Unicode im Text, Metadaten/C2PA in der Hülle, Muster im Stil
- ✓ Jede Bearbeitung mit **Vorher/Nachher-Lauf** belegen, nie blind vertrauen
- ✓ Ebene 3 hat **keinen zuverlässigen Selbst-Check** — Versprechen dazu sind unbelegbar
- ✓ **Nur eigene Inhalte** — Provenance fremder Dateien bleibt unangetastet

■ // BLEIB DRAN

Virale Repos, nüchtern geprüft.

20.000 Sterne sind ein Signal, kein Beleg. Ich schaue bei solchen Tools nach, was technisch dahintersteckt, bevor ich sie empfehle. Folg mir für die nächste Einordnung.

@ki.borish

Fakten verifiziert am 04.09.2026 aus dem Repository, der GitHub API und den Release-Notes von guillaumemeyer/watermarks-remover. Sterne-Zahlen sind ein Snapshot vom 04.09.2026, 16:30 UTC. Die Wirksamkeit auf statistische Wasserzeichen ist nicht unabhängig belegt. Dieser Guide ist eine technische Übersicht für eigene Dateien und keine Rechtsberatung — prüfe Kennzeichnungspflichten für deinen Fall selbst oder lass sie prüfen.