

# **Kartpullautin-Anleitung zur Vegetationskartierung**

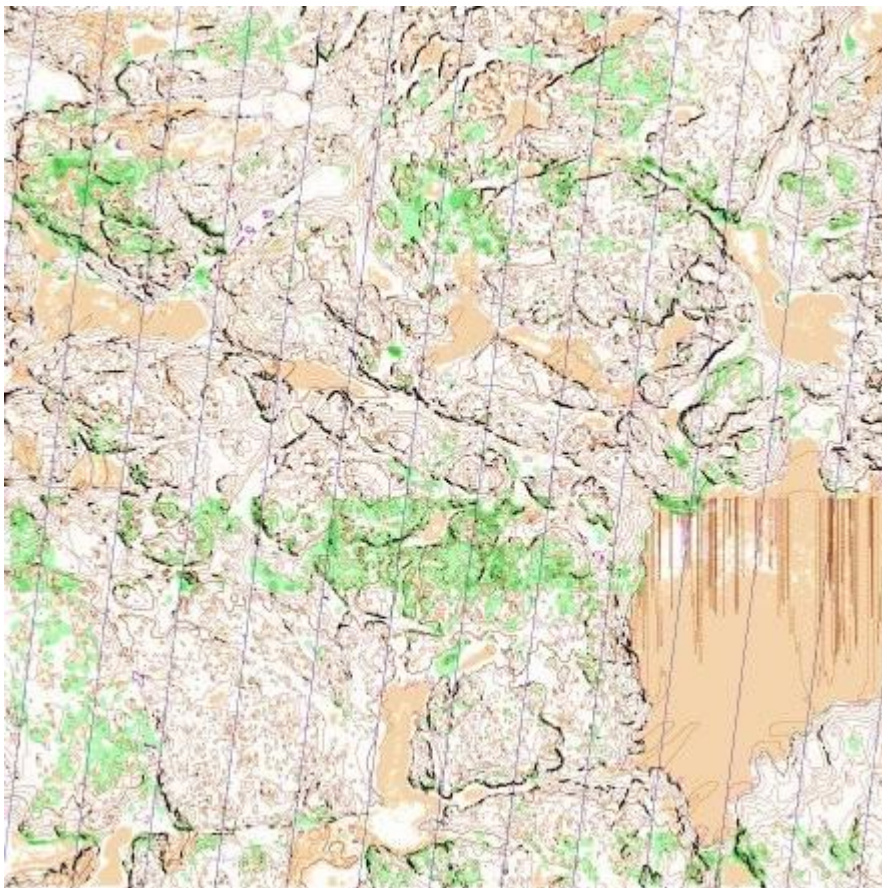
Anleitungsversion 20131027 / Jarkko R.

Deutsche Übersetzung: Domonkos Gy.

Für diese Aufgabe benötigen Sie ein Gebiet, in dem Laserdaten mit einer bestehenden Karte mit Grünkartierung überlappen oder Sie haben selbst die Vegetation kartiert oder verfügen über Bilder, die Sie zur Überprüfung der Grünkartierung verwenden können.

1. Öffnen Sie `\pullauta.ini` und setzen Sie den Parameter `\greenshades` auf einen breiten Bereich, um ausreichend Grün für den Abgleich zu erhalten. Beispiel:  
`greenshades=0.4|0.8|1.3|1.9|2.6`
2. Erstellen Sie die Karte wie gewohnt. Öffnen Sie `\pullautus.png` und überprüfen Sie die Grünkartierung. Falls grüne "Streifen" auftreten, also lineare Bereiche, die grüner sind als andere, müssen Sie diese durch Anpassung des Wertes `\pointvolumefactor` in der `.ini`-Datei ausgleichen. Ändern Sie beispielsweise von `\0.15` auf `\0.35` und erstellen Sie die Karte erneut. Wiederholen Sie diesen Schritt, bis das Grün ausgeglichen wirkt. Es kann hilfreich sein, dies einmal mit einem zu niedrigen und einmal mit einem zu hohen Wert zu testen und dann einen Wert dazwischen zu schätzen. Normalerweise sollte `\pointvolumefactor` zwischen `\0` und `\0.5` liegen.

Um die Iteration zu beschleunigen, können Sie nur den Vegetationskartierungsteil erstellen, wenn Sie die vollständige Karte einmal erstellt haben. Mit dem Befehlszeilenbefehl `\pullauta makevegenew` wird die Vegetationskartierung erstellt, und dann wird mit dem Befehl `\pullauta` die PNG-Karten mit dieser neuen Vegetation gerendert.



*Bild 1. Typische grüne Streifen, Bereiche, in denen sich Scanlinien überlappen und die Punktdichte höher ist.*

3. Wenn der Abgleich stimmt, schneiden Sie einen kleinen Teil der Daten (einen Bereich mit verschiedenen Grüntönen, die auf der O-Karte kartiert sind) mit dem Lastools-Befehl aus:

```
las2las -i in.laz -o out.laz -clip 630250 4834500 630500 4834750
```

Die Koordinaten müssen Sie selbst herausfinden. Sie können dies auch mit der gesamten Kachel durchführen, es dauert nur etwas länger.

Erstellen Sie dann die Karte erneut, indem Sie die `.laz`-Datei auf `pullauta.exe` ziehen. Ziel ist es, gute Werte für den Parameter `greenshades` zu finden. Beispiel:

```
greenshades=0.4/0.8/1.3/1.9/2.6
```

Der erste Wert ist der Schwellenwert für das hellste Grün, der letzte für das dunkelste (fünfte) Grün.

4. Vergleichen Sie dann die Grünkartierung der PNG-Karte mit dem Grün der echten O-Karte. Wenn beispielsweise das helle Grün der O-Karte zwischen dem hellsten und dem zweithellsten Grün (0.4 und 0.8) liegt, setzen Sie `greenshades` auf:

```
greenshades=0.4/0.5/0.6/0.7/0.8
```

und erstellen Sie die Karte erneut. Verwenden Sie die neue PNG-Kartenversion, um den richtigen Wert für helles Grün, z.B. `0.68`, zu ermitteln.

5. Wiederholen Sie die gleiche `greenshades`-Iteration, um das dunklere Grün richtig zu erhalten.
6. Wiederholen Sie die gleiche `greenshades`-Iteration, um das dunklere Grün richtig zu erhalten.

```
greenshades=0.68/1.13/99/99/99
```

Sie können dies ein drittes Mal wiederholen, um einen dritten Grünton zu erhalten, wenn Sie möchten.

Die Grundkonfiguration ist abgeschlossen, aber Sie sind möglicherweise noch nicht zufrieden damit. Niedriger Wald kann im Vergleich zu hohem Wald zu grün aussehen oder umgekehrt. Oder ein Teil, der dunkelgrün sein sollte, ist weiß, auch wenn ein anderer Teil des Gebiets gut kartiert ist.

## 7. Ausgleich von hohem und niedrigem Wald.

Es gibt eine Reihe von Werten in der `.ini`-Datei, um dies zu tun:

```
thresold1=1/2.5/0.39  
thresold2=2.5/6/0.43  
thresold3=6/10/0.44  
thresold4=10/20/0.41  
thresold5=20/99/0.38
```

Die ersten beiden Zahlen sind die Höhe des Walddachs. Wie Sie sehen können, sind alle Höhenbereiche zwischen 1 und 99 Metern definiert. Die dritte Zahl ist der Ausgleichswert. Je niedriger der Wert ist, desto einfacher wird der Wald mit dieser Dachhöhe grün. Sie können mit demselben Faktor für alle beginnen und dann den richtigen Wert einstellen.

Sie können so viele Zeilen haben, wie Sie möchten, und die Dachzonen (Schwellenwerte) nach Belieben festlegen, nummerieren Sie sie einfach von 1 bis n

## 8. Behebung des Problems "etwa 8 m hoher dichter Busch wird überhaupt nicht als grün kartiert."

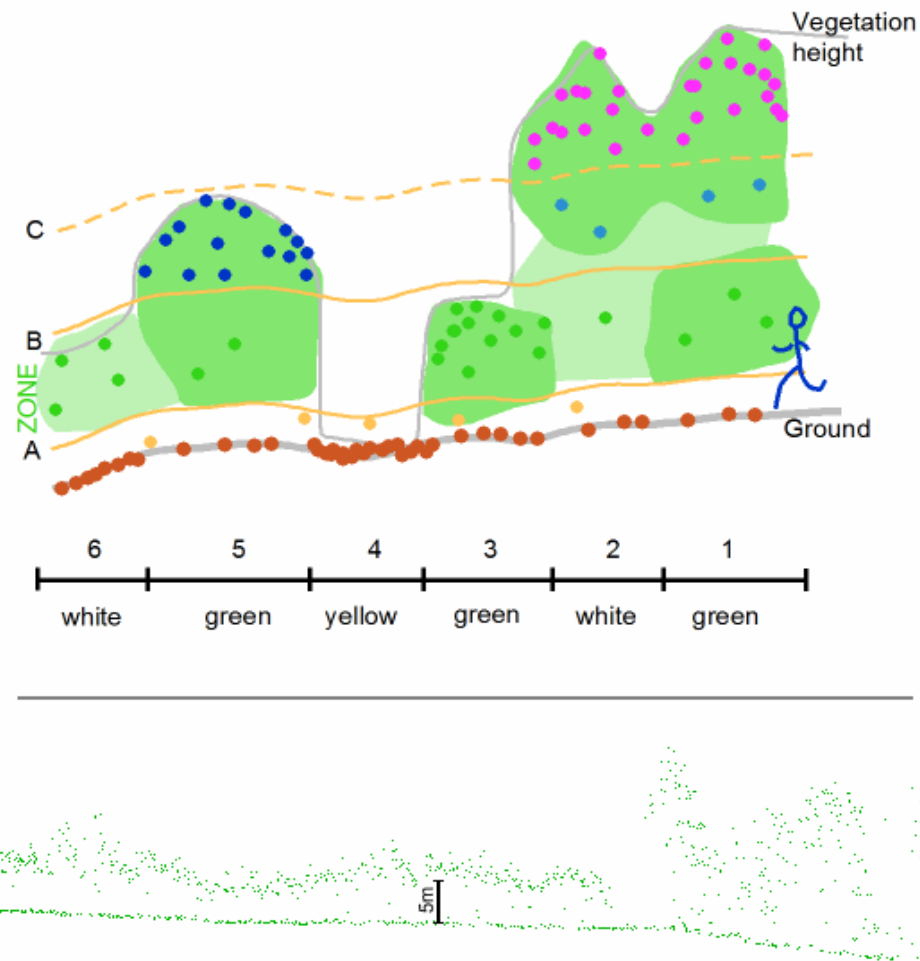
Wenn der Busch dicht ist, gibt es möglicherweise nicht viele Punkte zwischen Dach und Boden, weil das Lidar-System normalerweise eine ausreichend große Lücke zwischen den Punkten erwartet. Wenn er sehr dicht ist, gibt es möglicherweise nie eine ausreichend große Lücke, und es endet oft nur mit dem ersten Rücklauf bei 5..8 Metern und dem Boden. Und der Raum dazwischen sieht oft leer aus. Das ist ein Problem, denn wir interessieren uns hauptsächlich für Punkte etwa 0,5 .. 5 Meter über dem Boden, weil Treffer dort anzeigen, dass dort etwas ist und es einem im Weg sein könnte, wenn man versucht, dort zu laufen. Deshalb stellen wir Pullautin normalerweise so ein, dass es Punkte zwischen etwa 0,8 und 5 Metern zählt. Pullautin vergleicht diese Zahl mit der Zahl der Punkte unterhalb des Parameters `greenground`, z.B. unterhalb von 0,8 Metern. Ein größeres Verhältnis von "grünen Treffern" bedeutet, dass es grüner ist. Um dieses Problem der leeren Zone zu lösen, hat Pullautin eine Reihe von Parametern, um zu definieren, wie grüne Treffer berechnet werden:

$zone1=0.8/5/99/1$   
 $zone2=5/9/11.0/0.75$

Erklärung:

- `zone1`: Treffer zwischen 0,8 und 5 Metern über dem Boden werden als 1 Treffer gezählt, wenn das Walddach niedriger als 99 Meter ist.
- `zone2`: Wenn der Punkt noch nicht berechnet wurde, dann, wenn der Treffer zwischen 5 und 9 Metern über dem Boden liegt und das Walddach niedriger als 11 Meter ist, wird er mitgezählt, aber nicht als ein Treffer, sondern nur als 0,75 Treffer.

Auf diese Weise können Sie Daten aus verschiedenen Höhen sammeln und abschätzen, was aus Sicht des Läufers am wahrscheinlichsten ist. Sie müssen also wissen, wie Ihr Wald aussieht. Wenn Sie hauptsächlich 8 m hohe Kiefern ohne tiefe Äste haben (z.B. Barbate-Kiefernwald), sollten Sie Treffer bei 7 Metern überhaupt nicht zählen. Wenn Ihre 8 m hohen Bäume jedoch durchgehend dicht sind, sollten Sie höhere Treffer zumindest teilweise mitzählen, wenn das Dach unter etwa 12 Metern liegt.



*Bild 2.*

Bild 2 ist eine leicht veraltete Version der Pullautin-Grünkartierung, gibt aber dennoch eine Vorstellung davon, wie die Berechnung durchgeführt wird. Braune und gelbe Punkte werden als Boden berechnet, und diese Zahl wird mit der Anzahl der grünen Treffer verglichen. Grüne Punkte werden als grün gezählt, dunkelgrüne Punkte teilweise, hellblaue und violette Punkte werden nicht gezählt.

Nachdem Sie die Art und Weise, wie Punkte gezählt werden, geändert haben, müssen Sie wahrscheinlich iterieren und die Werte für den Parameter `greenshades` erneut bestimmen.

## Andere Parameter:

- "**greenround**“: Mit diesem Parameter setzen Sie den Schwellenwert, der verwendet wird, um Treffer in diese beiden Gruppen (grün und unter grün) zu unterteilen. Pullautin vergleicht und berechnet den Grünwert, der mit den Werten `pointvolumefactor`, `threshold` und schließlich mit `greenshades` abgeglichen wird, um den Grünton für das Gebiet festzulegen.
- "**blocksize**“: Der Bereich, der für die Berechnung verwendet wird, wird mit `blocksize` \* 3m x 3m durchgeführt. Ein kleinerer Block ergibt eine stärker verstreute (aber genauere Lage der Kanten) Grünfärbung, ein größerer Block ergibt eine gleichmäßigere und zuverlässigere Farbe, aber die Lage der Kanten ist möglicherweise nicht so genau
- "**dotsize**" ist der Durchmesser, wie groß ein Punkt auf der Karte für jeden `blocksize` gezeichnet wird.
- "**yellowheight**": Niedriger als dieser Wert wird als offen/gelb angesehen.
- "**yellowthreshold**": Wie groß der Anteil der Treffer unterhalb von `yellowheight` sein muss, um als gelb kartiert zu werden. Normalerweise etwas wie 0,88 (über 88% der Punkte sollten unterhalb von `yellowheight` liegen)

**Problem Laub- vs. Nadelbäume:** Normalerweise wird das Scannen durchgeführt, wenn keine Blätter vorhanden sind. Dies kann dazu führen, dass immergrüne Teile des Waldes grüner aussehen als der Rest des Gebiets. Pullautin hat keine Möglichkeit, dieses Problem zu beheben, außer zwei Versionen der Grünkartierung zu erstellen und sie manuell mit einem Fotobearbeitungsprogramm unter Verwendung von Luftbildern zusammenzuführen, um zu bestimmen, was am wahrscheinlichsten ist. Normalerweise ist dies jedoch kein großes Problem.

# Inhalt von pullauta.ini auf Deutsch

```
#-----#
# Parameter für den Karttapullautin-Pullautus-Prozess #
#-----#

##### PARAMETER #####
# Experimentelle Unterholzparameter. Kleinere Werte erzeugen mehr Unterholzstreifen.
# Normales Unterholz
undergrowth=0.35

# Unterholz-Weg
undergrowth2=0.56

# Hinweis: Sie müssen dies iterieren, wenn Sie diesen Modus verwenden. Mit den Befehlen 'pullauta
makevege' und dann 'pullauta' können Sie nur diesen Teil erneut verarbeiten.
# Höhe für Treffer unterhalb von Grün. Für die Grünkartierung werden Treffer unterhalb dieses
Wertes als Punkte berechnet, die durch Vegetation ~ Boden gegangen sind.
greenground=0.9
greenhigh=2
topweight=0.80
vegezoffset=0
greendetectedsize=3

### Hier berechnen wir Punkte. Wir können Höhenzonen und Faktoren für Grün verwenden. Beispiel:
# niedrig|hoch|Dach|Faktor
# zone1=1|5|99|1 # Punkte 1 bis 5 Meter werden als ein Treffer berechnet, wenn die höchsten
Bäume dort niedriger als 99 Meter sind
# zone2=5|9|11.0|0.75 # zusätzlich werden Punkte 5 bis 9 Meter als 0,75 Punkte gewertet, wenn die
höchsten Bäume niedriger als 11 Meter sind.
# Es können so viele Zonen wie gewünscht vorhanden sein

# niedrig|hoch|Dach|Faktor
zone1=1.0|2.65|99|1
zone2=2.65|3.4|99|0.1
zone3=3.4|5.5|8|0.2

## Hier bestimmen wir, wie empfindlich wir Grün für verschiedene (hohe oder niedrige) Waldtypen
erhalten.
# Wenn beispielsweise ein hoher Wald mit großen Bäumen im Vergleich zu einem niedrigen Wald zu
grün wird, können wir dies hier anpassen.
# Dach niedrig|Dach hoch| Verhältnis Grüntreffer/Boden, um Grünfaktor 1 auszulösen
thresold1=0.20|3|0.1
thresold2=3|4|0.1
thresold3=4|7|0.1
thresold4=7|20|0.1
thresold5=20|99|0.1

## In Bereichen, in denen sich Scanlinien überlappen, haben wir eine zwei- oder dreifach höhere
Punktdichte. Dies kann diese Bereiche grüner oder weniger grün erscheinen lassen. Verwenden Sie
diese Parameter, um dies auszugleichen.
# Die Formel lautet: * (1-pointvolumefactor * mydensity/averagedensity) ^ pointvolumexponent
# pointvolumefactor = 0 bedeutet also kein Ausgleich/Effekt

pointvolumefactor=0.1
pointvolumexponent=1

# Grün-Gewichtung, wenn der Punkt der einzige Rücklauf ist - dies sind normalerweise Felsbrocken
oder ähnliches
# Diese werden daher nur teilweise gezählt
firstandlastreturnfactor=1

# Grün-Gewichtung für den letzten Rücklauf - diese können Vegetation sein, aber weniger
wahrscheinlich als frühere Rückläufe
lastreturnfactor=1

firstandlastreturnasground=3
# Grünwerte zum Auslösen von Grünschattierungen. Verwenden Sie eine hohe Zahl wie 99, um einige
der Schattierungen zu vermeiden.
```

```

#greenshades=0.0|0.1|0.2|0.3|0.4|0.5|0.6|0.7|0.8|0.9|1.0|1.1|1.2|1.3|1.4|1.5|1.6|1.7|1.8|1.9|2.0|
2.1|2.2|2.3|2.4|2.5|2.6|2.7|2.8|2.9|3.0

greenshades=0.2|0.35|0.5|0.7|1.3|2.6|4|99|99|99|99

# Farbton für das hellste Grün. 255 ist weiß.
lightgreentone=200

# Nicht ändern
greendotsize=0

# Blockgröße zur Berechnung des Verhältnisses von Treffern unterhalb von Grün. Verwenden Sie 3,
wenn greendetectedsize kleiner als 5 ist, wenn
# es größer ist, verwenden Sie 1
groundboxsize=1

# Medianfilterung des grünen Rasterbildes. Zwei Runden
# Verwenden Sie 1, um keine Filterung durchzuführen.
medianboxsize=9
medianboxsize2=1

## Gelb-Parameter
### Treffer unterhalb dieses Wertes werden als gelb berechnet
yellowheight=0.9

### Wie groß der Anteil der Punkte unterhalb von yellowheight sein muss, um Gelb auszulösen
yellowthreshold=0.9

#####
## Klippenmacher Mindesthöhenwerte für jeden Klippentyp. Vertikaler Abfall pro 1 Meter
horizontaler Entfernung
## cliff1 = Diese Klippen werden gelöscht, wenn die Steilheit größer als der Wert unten ist
## cliff2 = unpassierbare Klippe

cliff1=1.15
cliff2=2.0
cliffthin=1

cliffsteepfactor=0.38
cliffflatplace=3.5
cliffnosmallcliffs=5.5

cliffdebug=0
## Drehwinkel der Nordlinien (im Uhrzeigersinn) und Breite. Breite 0 bedeutet keine Nordlinien.
northlinesangle=0
northlineswidth=0

## Formlinienmodus, Optionen:
# 0 = 2,5m Intervall, keine Formlinien
# 1 = 2,5m Intervall, jede zweite Kontur dünn/dick
# 2 = 5m Intervall, mit einigen gestrichelten Formlinien dazwischen, falls erforderlich

formline=2

# Steilheitsparameter für Formlinien. Ein größerer Wert ergibt mehr und ein kleinerer Wert
weniger Formlinien.
formlinesteeptness=0.37

## Zusätzliche Länge der Formlinien an den Scheitelpunkten
formlineaddition=17

## Kürzeste Lücke zwischen den Enden der Formlinien an den Scheitelpunkten
minimumgap=30

# Strich- und Lückenparameter für Formlinien
dashlength=60
gaplength=12

# Intervall für Indexkonturen. Wird nur verwendet, wenn der Formlinienmodus 0 ist
indexcontours=12.5

# Konturen glätten. Ein größerer Wert glättet die Konturen mehr. Standard = 1. Werte zwischen
etwa 0,5 und 3,0 ausprobieren.

```

```
smoothing=0.7

# Krümmung. Wie kurvig die Konturen erscheinen. Standard = 1. Ein größerer Wert erzeugt
kurvigere/übertriebener Kurven (Einbuchtungen und Vorsprünge)
curviness=1.1

# Hügelqualifikation. Standard = 0,8. Bereich 0,0 ... 1,0. Größere Werte ergeben weniger, aber
deutlichere Hügel.
knolls=0.6

# xyz-Faktoren, z.B. für die Umrechnung von Fuß in Meter
coordxfactor=1
coordfactor=1
coordzfactor=1

# Ausdünnungsfaktor von las/laz zu xyz. Beispiel: 0,25 lässt 25% der Punkte übrig
thinfactor=1

# Wenn Punkte als Wasser klassifiziert sind, wird diese Klasse blau gezeichnet (Kommentar
entfernen, um dies zu aktivieren)
# waterclass=9

# Wasserhöhe, Höhe unterhalb dieses Wertes wird blau gezeichnet (Kommentar entfernen, um dies zu
aktivieren)
# waterelevation=0.15

# Wenn Punkte als Gebäude klassifiziert sind, wird diese Klasse schwarz gezeichnet (Kommentar
entfernen, um dies zu aktivieren)
# buildingsclass=6

# Gebäudeerkennung. 1=ein, 0=aus. Diese werden als lila mit schwarzen Rändern gezeichnet.
Hochgradig experimentell.
detectbuildings=0

# Stapelverarbeitungsmodus, verarbeitet alle laz- und las-Dateien dieses Verzeichnisses
# aus=0, ein=1
batch=0

# Prozesse
processes=2

# Ausgabeordner für Stapelverarbeitung
batchoutfolder=./out

# Eingabeordner für Stapelverarbeitung
lazfolder=./in

# Wenn relative Pfade nicht funktionieren, versuchen Sie absolute Pfade wie
c:/yourfolder/lasfiles

# Karttappullautin kann Vektordateien rendern.
# Maastotietokanta des National Land Survey of Finland benötigt keine Konfigurationsdatei. Für
das Rendern dieser lassen Sie diesen Parameter leer.
# Für andere Datensätze wie Fastighetskartan von Lantmateriet (Schweden) wird eine
Konfigurationsdatei benötigt.

vectorconf=
# vectorconf=osm.txt
# vectorconf=fastighetskartan.txt

# Vektordateien sollten sich in ZIP-Dateien befinden und im Stapelverarbeitungs-Eingabeordner
platziert oder per Drag-and-Drop auf pullauta.exe gezogen werden

# maastotietokanta, diese Ebenen nicht rendern, kommagetrennt
mtkskiplayers=

# Kommentar entfernen, um keine Siedlungsfarbe zu verwenden (diese Ebenen überspringen, die
Pullautin normalerweise olivgrün zeichnet)
# mtkskiplayers=32000,40200,62100,32410,32411,32412,32413,32414,32415,32416,32417,32418

# Farbe für Vektorgebäude (RGB-Wert 0,0,0 ist schwarz und 255,255,255 ist weiß)
buildingcolor=0,0,0
```

```

# Im Stapelmodus, werden auch einige temporäre Dateien in den Ausgabeordner kopiert und
# zugeschnitten?
# Ordner. 1=ein 0=aus. Verwenden Sie dies, wenn Sie Vektorhöhenlinien und ähnliches für jede
# Kachel verwenden möchten.

savetempfiles=0

# Im Stapelmodus wird das gesamte temporäre Verzeichnis so gespeichert, wie es ist
savetempfolders=0

# Das Intervall der zusätzlichen dxf-Höhenlinienschnitt (roh, für die Kartierung). 0 =
# deaktiviert. Der Wert 1,125 ergibt solche Intervallhöhenlinien
# basemapinterval=0

#####
# Einstellungen speziell für die Rust-Version von Kartapula, Standardwerte sollen der
# ursprünglichen Perl-Version von Kartapula entsprechen #
#####
# contour_interval legt das Höhenlinienintervall in Metern für die Ausgabekarte fest
contour_interval=5

# depression_length legt die maximale Länge der zu markierenden Senken fest. Der Originalwert der
# Perl-Version ist fest auf 181 codiert.
# Setzen Sie eine sehr große Zahl, wenn alle Senken markiert werden sollen.
depression_length=181

# yellow_smoothing auf 1 setzen, um einen Glättungseffekt auf die gelben Bereiche anzuwenden, der
# der Glättung der grünen Bereiche entspricht
yellow_smoothing=0

# Wenn yellow_smoothing=0, dann ist yellowmedianboxsize eine andere Möglichkeit, Gelb mit einem
# Medianfilter dieses Wertes zu glätten
yellowmedianboxsize=0

# vege_bitmode auf 1 setzen, um ein Bit auszugeben
vege_bitmode=0

# label_formlines_depressions auf 1 setzen, um den Senken in der Formlinien-Vektordatei eine
# separate Beschriftung hinzuzufügen
label_formlines_depressions=0

# Einige Höhenlinien entfernen, wenn es zu steil ist, um deutliche Höhenlinien zu zeichnen
remove_touching_contours=0

# vegeonly auf 1 setzen, um nur die vegetationsbezogenen Dateien zu generieren und den Rest zu
# überspringen
vegeonly=0
# contouronly auf 1 setzen, um nur die höhenlinienbezogenen Dateien zu generieren und den Rest
# zu überspringen
contouronly=0
# cliffonly auf 1 setzen, um nur die klippenbezogenen Dateien zu generieren und den Rest zu
# überspringen
cliffonly=0
# Nur eines von vegeonly, contouronly und cliffonly kann gleichzeitig gesetzt werden

#-----#
#           EXPERIMENTELLE OPTIONEN           #
#           (Keine Stabilitätsgarantien)      #
#-----#

# Setzen Sie experimental_use_in_memory_fs auf 1, um ein speicherinternes Dateisystem für alle
# temporären Dateien zu verwenden.
# Dies ist nützlich, wenn die Dateien klein genug sind, um in den Speicher zu passen, da es die
# benötigte E/A-Menge reduziert, auf Kosten der Verwendung von mehr RAM.
experimental_use_in_memory_fs=0

# Experimentelle Parameter. Ändern Sie diese nicht, es sei denn, Sie möchten experimentieren
scalefactor=1
zoffset=0
#skipknolldetection=0

```