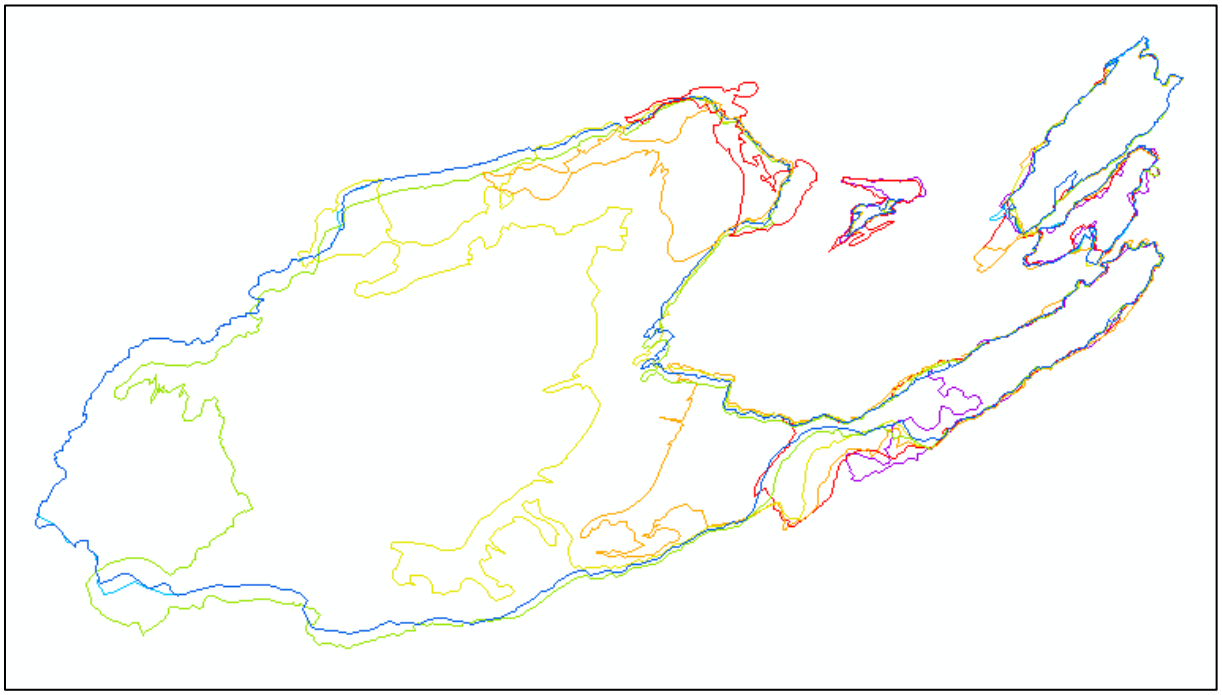
Converting multiple KML files into one SHP file at once

in order to track temporal evolution of glacial lakes



Multiple outlines of Gaulisee at Gauligletscher from 1980 to 2020

Supervision

Prof. Dr. Andreas Zischg

Geodata Analysis and Modelling

Institute of Geography

University of Bern

Marianne Greber

marianne.greber@students.unibe.ch

15-105-588

27.01.2022

1. Introduction

During the seminar “Geodata Analysis and Modelling” geography students are given the opportunity to take their first steps in programming with python and sharing code through the GitHub platform. The new skills can either be used to write a script for a project topic suggested by the supervisor or student or as part of the students masterthesis.

The code and results presented here have been part of my master thesis “Characteristics and high-resolution spatio-temporal evolution of glacial lakes in the Rhine catchment (Swiss Alps) since 1980”. For the master thesis, orthophotos of the Rhine catchment from 1980 until 2020 have been analysed and each glacial lake exceeding a lake area of 1000 m^2 has been mapped. In order to track the temporal evolution, the lake outlines have been mapped for multiple timesteps. Both, the orthophotos and the mapping tool are available and provided on the Swiss Geoportal (swisstopo 2021). Each lake outline was exported from the Swiss Geoportal as a polygon feature. In total, 300 glacial lakes were detected in the Rhine catchment and 1882 lake polygons were created.

The Keyhole Markup Language (KML) is a format often used to store geographic information in .kml-files and .kmz-files. One of the main reasons is, that .kml or .kmz-files can be shared easily on the internet since it contains all elements of a data layer or map (Esri 2022). The created glacial lake polygons on the Swiss Geoportal are also saved in the KML format (.kml) and can be read by free applications such as the Swiss Geoportal, ArcGIS Explorer or Google Earth. However, it can only be read by ArcMap, if it is converted into a layer file first. The conversion tool “KML To Layer” in Arcmap allows to do that for a .kml or kmz-file one by one but not for multiple files at once.

The code presented here was written to convert multiple .kml or .kmz-files (input data) into one single feature class saved in a shapefile (.shp), that can be read and edited in Esri ArcMap. Furthermore, each feature is labelled with the name of the original .kml or .kmz-file in order to distinguish the different input data in the resulting shapefile and to join an excel table to the attribute table later on.

2. Code

The code was written in PyCharm Community Edition 2021.1.1 using Python 2.7. The code is structured in four steps and split into two separate files. The first file “1_Converting_KML2SHP” contains steps one and two and the second file “2_Merging_SHP” contains step three and four. The four steps are as follows:

1) Converting

   In this step each .kml or .kmz-file is converted into a layer, based on the ArcMap tool “KML To Layer”. While looping through the .kml or .kmz-file, the code prints “converting: filename” for each file in order to keep track of progress. The statement “FINISHED CONVERTING” marks the end of this step.

2) Copying

   In this step, each created feature class is copied into the master geodatabase and named as the original .kml or .kmz-file. While looping through the feature classes, the code prints “copying: filename” for each feature class in order to keep track of progress. The statement “FINISHED COPYING” marks the end of this step.

3) Creating attribute field

   In this step, an attribute field is added to each feature class and filled with the name of the feature class. While looping through the feature classes, the code prints “filling attribute field: filename” for each feature class in order to keep track of progress. The statement “FINISHED ATTRIBUTES” marks the end of this step.

4) Merging

   In this final step, all feature classes are merged into one single feature class. The statement “FINISHED MERGING” marks the end of this step.

While writing the code, it has continuously been tested with a small data sample. Once it was completed, the amount of data was increased and it seemed as if the execution time increased disproportionately much. Therefore, a timer has been added to each file. Indeed, multiple runs of the code showed that the execution gets slower when handling big amounts of data (>500 .kml-files). However, it also seemed as if the problem was mainly in step one where the input .kml-files are converted into feature classes. Since it is not necessary for this step to process all the input data at once, the code was split into two separate files. Thus, file “1_Converting_KML2SHP” is converting the .kml-files into feature classes and can be run multiple times for multiple parts of the input data. Afterwards, file “2_Merging_SHP” can be run once on the entire data since execution time of step three and four seemed increase proportionally with the amount of data.

However, it is clear the despite the automated conversion of KML to SHP saves a lot of manual work, the execution time of the code is still very long. The code should be improved in that regard if it is to be applied to more than 1000 .kml-files at once.

3. Results

As mentioned before, the code was applied in the master thesis thesis “Characteristics and high-resolution spatio-temporal evolution of glacial lakes in the Rhine catchment (Swiss Alps) since 1980”. Glacial lake outlines have been mapped from orthophotos acquired between 1977 and 2020. Each glacial lake outline was mapped from all the available orthophotos during the observation period and each outline was saved in a KML format that had to be converted.

The observation period from 1980 to 2020 has been divided into seven glacial lake inventory (GLI) timesteps, that together make a multi-temporal glacial lake inventory:

Inventory timestep	Assigned period
GLI1980	1977 – 1985
GLI1990	1986 – 1994
GLI2000	1995 – 2002
GLI2005	2003 – 2007
GLI2010	2008 – 2012
GLI2015	2013 – 2017
* The most recent available data at the time of mapping are from 2020.	

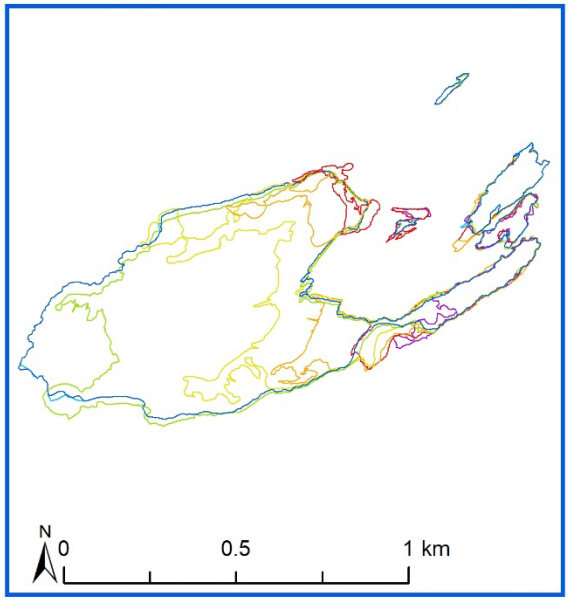
The map shows the compiled multi-temporal glacial lake inventory for three glacial lakes: the Gaulisee, the Hüfisee and an unnamed lake at the Claridengletscher. The temporal evolution of each lake is shown by colouring each GLI timestep differently.

The unnamed lake at Claridengletscher (canton Uri) (c) is an example for a typical formation of a glacial lake. In GLI1980 (purple) only a very small water pond at the north eastern exists. In GLI1990, there are already two separate and larger water ponds visible on the orthophotos, divided by the glacier tongue. As the glacier continued to retreat, more space became available and the two ponds merge to one glacial lake that grows in size until GLI2020.

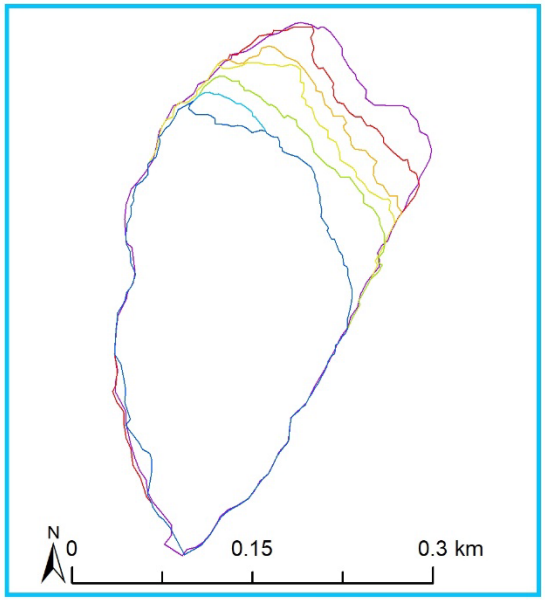
Hüfisee at Hüfifirn (canton Uri) (b) was mapped the first time in 1963 on topographic maps. Its formation and growth period finished before the observation period started around 1980. However, the outlines of Hüfisee still show a very interesting evolution. The lake area is largest in GLI 1980 and reduces continuously until GLI2020. Through observation of the orthophotos, the loss of lake area can be attributed to the sediment input in the lake and the growing delta at the inflow.

Temporal evolution of glacial lakes

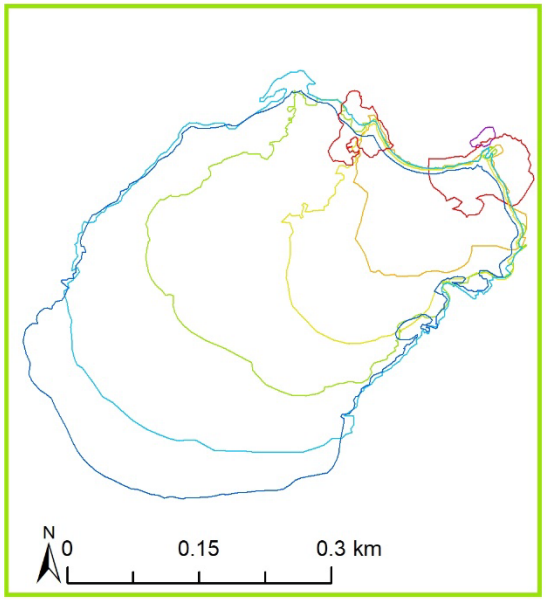
a) Gaulisee at Gauligletscher

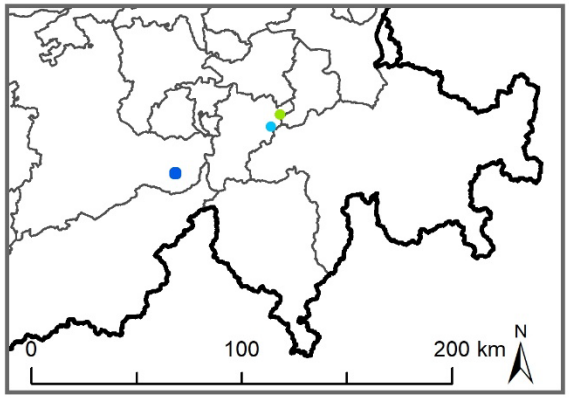


b) Hüfisee at Hüfifirn



c) unnamed glacial lake at Claridengletscher





GLI1980
GLI1990
GLI2000
GLI2005
GLI2010
GLI2015
GLI2020

national boundaries
cantonal boundaries

Source
Glacial lake inventory GLI: Marianne Greber
Boundaries national/cantonal: swisstopo
Author: Marianne Greber
Date: January 2022

Coordinate system: CH1903+ LV95
Projection: Hotine Oblique Mercator Azimuth Center
Date: CH1903+

On first glance, the temporal evolution of Gaulisee at Gauligletscher (canton Bern) (c) is not clearly distinguishable as for the other two lakes. At first in GLI1980, only the three smaller lake areas on the eastern border of the map extent are visible and the main lake area is still occupied by the glacier. A closer look at the aerial images shows, that the three basins are on different elevations, trapping the meltwater on three different levels. As the glacier continues to retreat, the topmost pond can expand and at the same time another pond forms on the northern side of the glacier tongue. Afterwards, the lake evolution is very similar to c), the two ponds merge and form one single lake polygon that grows in size over time. Again, GLI2020 marks the largest mapped glacial lake extent.

4. Conclusion

Even though the code should be revised to improve the execution time, it nevertheless saved a lot of time and manual work to convert the 1882 glacial lake polygons into one single shapefile. Additionally, execution time was reduced by splitting the code into two separate python files to process smaller parts of the input data at once in the first file (where the conversion happens) and the second file, where all the converted data is merged into one shapefile. Adding attribute fields with the original name of the .kml-files was crucial since it allowed distinguishing each glacial lake outline in a simple way and was needed to join further attributes from an excel table.

Overall, the seminar “Geodata Analysis and Modelling” gave a very nice introduction into programming in geography and very useful later, when I had to write a script to automate the conversion of the mapped glacial lake outlines. Without the script, I would not have been able to compile the inventory let alone analyse the spatio-temporal evolution of glacial lakes in the Rhine catchment.

References

Esri (2022) What is KML? https://desktop.arcgis.com/en/arcmap/latest/manage-data/kml/what-is-kml-.htm. Accessed 26 Jan 2022

swisstopo (2021) Swiss Geoportal. In: geo.admin.ch. https://map.geo.admin.ch. Accessed 1 Mar 2021