GIS-5 DEM Generation

2025-04-17

Per Halvorsen
pmhalvor@uio.no
GEO4460

1 Introduction

In this report, we will look into Domain Elevation Modeling (DEM). Specifically, we'll generate DEMs using the following techniques:

- Natural Neighbor Interpolation
- TIN (triangulated irregular network)
- Topo to Raster (Hutchinson's ANUDEM algorithm)

The goal of this exercise is to learn how to generate DEMs, see how more informative data can improve DEM estimation, and combine both quantative and visual analysis of the results.

2 Data

The data used to build this analysis was extracted from the N50 Dataset, by Kartverket hosted on GeoNorge.

The software used to generate this analysis was:

- Whitebox-tools: an open source Pytohn library for geospatial analysis
- Rasterio: a Python library for reading and writing geospatial raster data
- GRASS GIS: an open source GIS software w/ a Python API and additional capabilities including GUI & Stream Burning
- ArcGIS Pro: for the ANUDEM algorithm and to visual quality assessments

2.1 Preprocessing

Before we can go through the steps outline in the project description, we needed to convert the raw N50 data to a similar geodatabase as our example data, containing feature layers `contour_arc`, `elevation_point`, `lake_polygon`, and `river_arc`.

To do so we used ArcGIS Pro to run the following steps: 1. Import the N50 data into a new geodatabase. 1. Load the land cover data, elevation points, and contour lines to current map. 1. Create a feature layer containing a single polygon (retangle) around the area of interest. We chose a region in Rogaland around Preikestolen. 1. Clip the N50 layers to the rectangle polygon. 1. Convert labels in the land cover data to new feature layers polygons when objects type was `innsjø`,

innsjø regulert, or elv. 1. Convert river feature layers to polylines. 1. Rename relevant layers to contour_arc, elevation_point, lake_polygon, and river_arc. 1. Export the layers to a new geodatabase.

These steps were executed in ArcGIS Pro due to limitations of open-source tools for reading, parsing, and storing .gdb files. Also, doing these manually in a GUI ensured that we've selected an interesting region to analyze within the boundaries of our data, giving us a better foundation for understanding the results later.

Two more preprocessing steps were done directly in our Python workflow. These final preprocessing steps enforce the correct coordinate system across our layers and convert our contour lines to points, for more informative DEM interpolation. Generating points from contour dramatically improves the quality of the most simple DEMs, as will be discussed later.

3 Methods

3.1 DEM generation

For this specific analysis, we explored 3 interpolation methods: nearest neighbor, TIN gridding, and ANUDEM (Topo to Raster in ArcGIS Pro). The first two methods were automated and implemented in Python using Whitebox-tools and Rasterio; the last method was manually implemented in ArcGIS Pro.

First, one DEM was generated for each of the automated methods using only the contour points. Then, another DEM using only the *elevation* points was generated for each of the automated methods. This was deemed possible after recognizing the expeected inputs for the two methods required only a point layer.

For the more complex ANUDEM method, we provided all the information included in the example data: contours, elevation points, lakes, and rivers (streams). The raster output from this method extended further than the intended area of interest, so a mask was applied to extract only the region we were investigating. During this extraction, we made sure to store the output to the same directory where our other data was, in order to include this ANUDEM in the quality assessment later.

An extra DEM was create as an attempt to approximate the ANUDEM method using the TIN method. This was achieved by merging the TIN raster with a stream burned raster, created from the river feature layer. The resulting layer gave results comparable to the TIN gridded DEM, which were still a step behind the ANUDEM method. We only generated this DEM using the contour points TIN grid DEM.

3.2 Quality assessment

The quality assessment of the DEMs was done using a combination of visual and quantitative analysis.

3.2.1 RSME

Quantitative analysis was done by calculating the RMSE (root mean square error) between the DEMs and the original elevation points. The RMSE was calculated using the following formula:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(z_i - z_{pred})^2}$$

where (z_i) is the original elevation point, (z_{pred}) is the predicted elevation from the DEM, and (n) is the number of points. RMSE was calculated for each of the DEMs generated using the elevation points. We could have done a similar analysis using the contour points, but that was skipped due to time constraints.

3.2.2 Hillshade

The first visual analysis was done by generating hillshade rasters from each of the DEMs. Hillshade is a technique used to create a shaded relief map, which can help visualize the terrain and identify features such as ridges, valleys, and slopes. Inputs are typically just the DEM, and parameters azimuth and altitude of the light source. In Whitebox-tools, default values are set to 315° and 30°, which were used for our analysis.

Hillshade was generated for each of the DEMs, and the results were visually analyzed in ArcGIS Pro.

3.2.3 Slope

Similar to hillshade, slope is another means of visualizing the terrain. Again, only the DEM is needed as input, though units and zfactor can be set if needed. We just used the default parameters for these.

Slope outputs were again loaded into ArcGIS Pro for visual analysis.

3.2.4 Profile analysis

The visual analysis was done by plotting the profiles along an arbitrary transect line. Whitebox-tools was used to generate the profile analysis, storing outputs from each DEM to an HTML file. In order to plot all the profiles into a single Matplotlib plot, we needed to use Beautiful Soup to parse the HTML files and extract the relevant data to arrays.

A transect line was created using the `create_line` function in Whitebox-tools, which creates a line between two points. Start and stop points here were automatically selected from the extent of the input data. An inset is defined as well, to ensure the transect line is not too close to the edges of the DEM, which could result in some artifacts in the profile.

4 Results

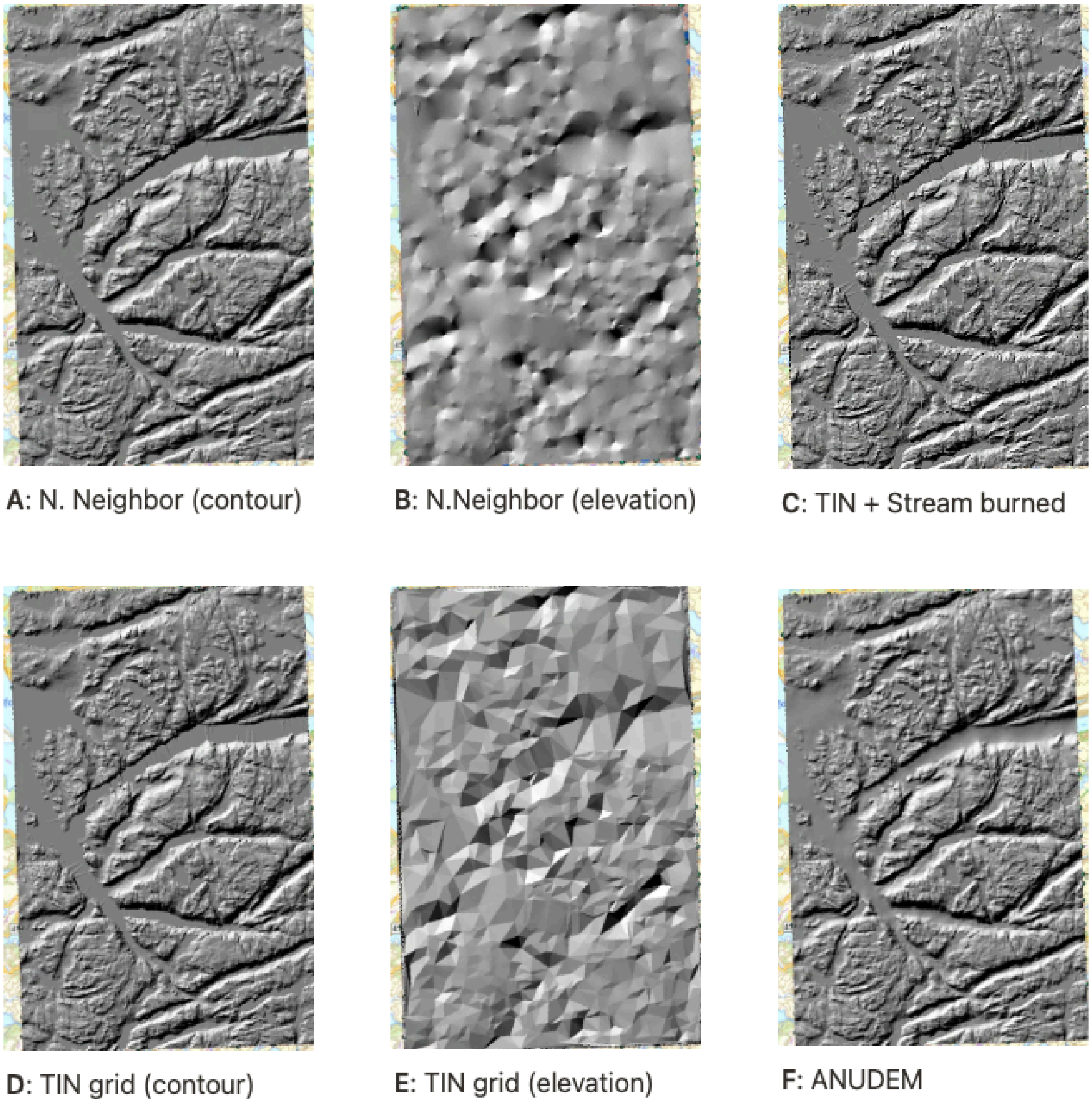
4.1 RSME results

DEM Type	RMSE	N Points
Natural Neighbor (Contour)	17.454156432215676	1212

DEM Type	RMSE	N Points
Natural Neighbor (Points)	2867.7097994172564	1212
TIN Gridding (Contour)	17.465622984616125	1212
TIN Gridding (Points)	2867.7112855837313	1212
Stream Burn (Contour TIN based)	17.478845283524066	1212
ANUDEM (ArcGIS Pro)	15.001615015426102	1212

Table 1: RMSE results for the DEMs generated using the elevation points.

4.2 Hillshade results



A: N. Neighbor (contour)

B: N.Neighbor (elevation)

C: TIN + Stream burned

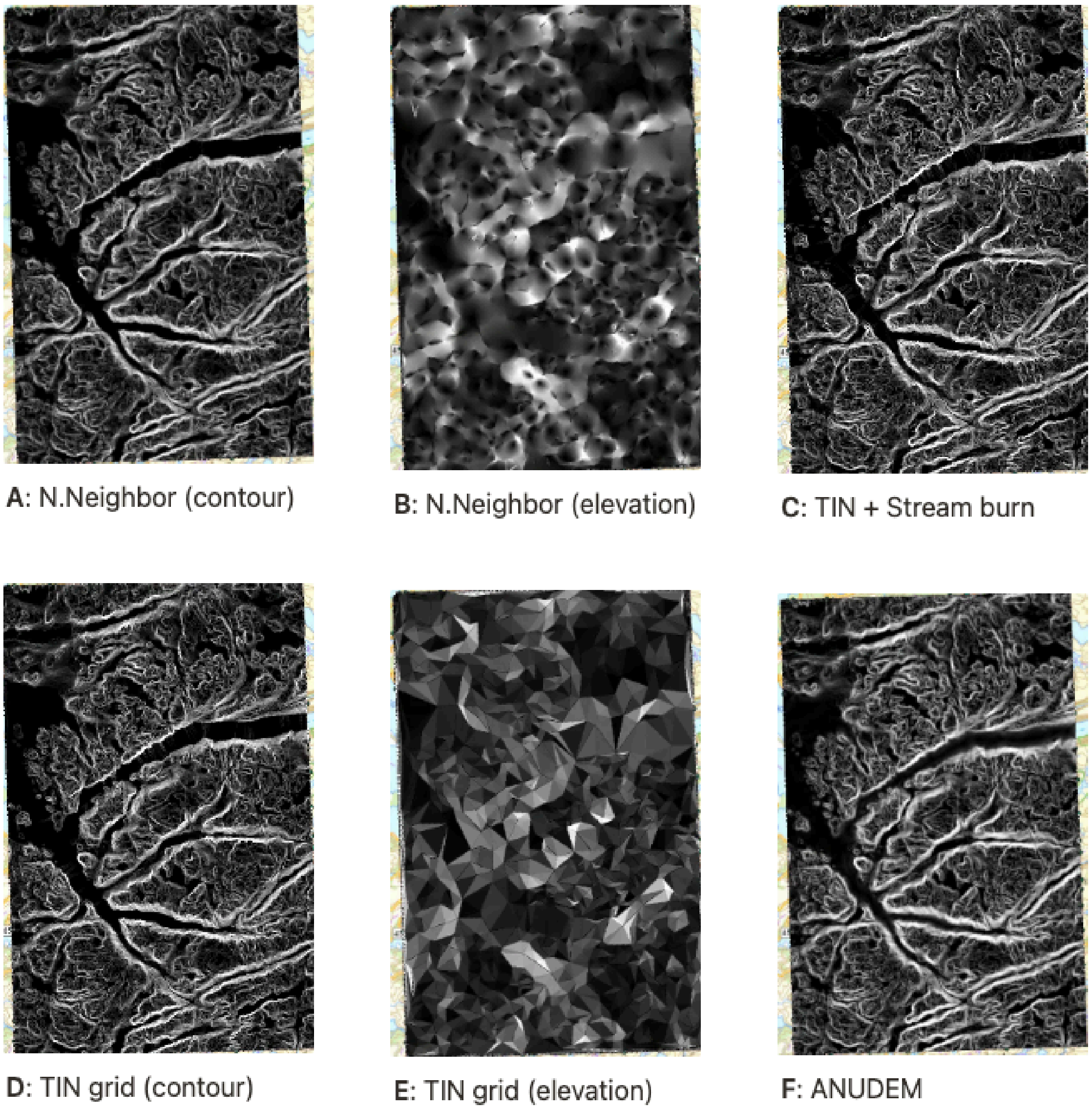
D: TIN grid (contour)

E: TIN grid (elevation)

F: ANUDEM

Figure 1: Hillshade results for the DEMs generated using the elevation points.

4.3 Slope results



A: N.Neighbor (contour)

B: N.Neighbor (elevation)

C: TIN + Stream burn

D: TIN grid (contour)

E: TIN grid (elevation)

F: ANUDEM

Figure 2: Slope results for the DEMs generated using the elevation points.

4.4 Profile results

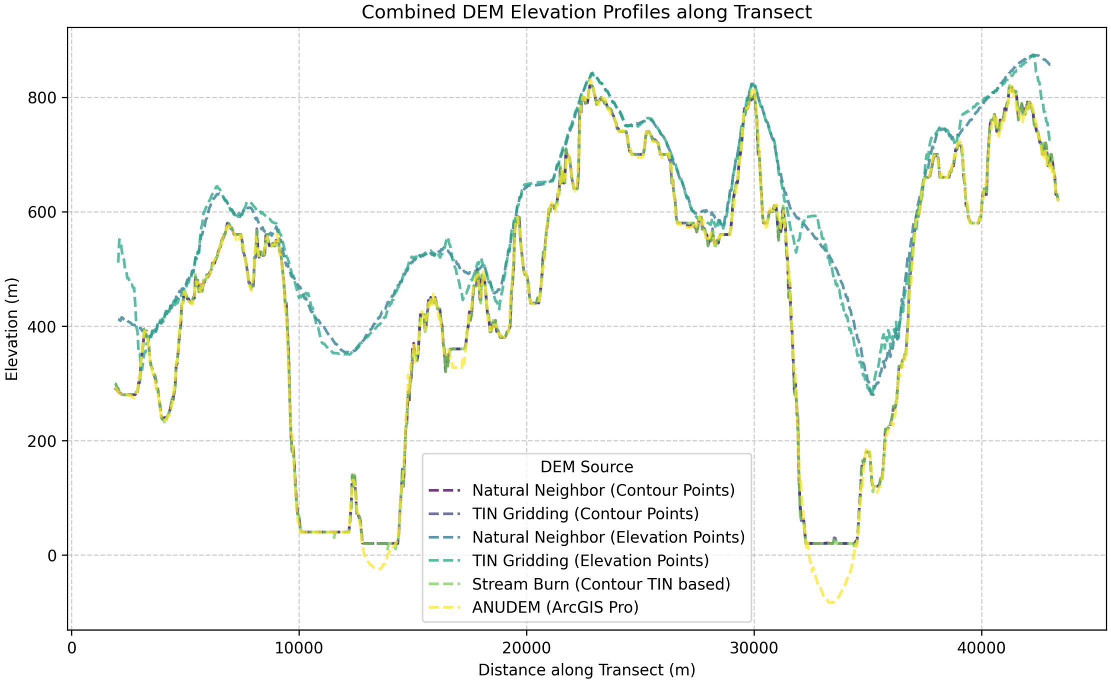


Figure 3: Profile results for the DEMs generated using the elevation points.

5 Discussion

The results from the RMSE analysis displayed in Table 1 show that the ANUDEM method performed best, with a RMSE of 15.00 m. Though, when considering the profile analysis, it become apparent how this method was able to beat out the automated methods. Look at the elevations *below zero* in Figure 3. It is only the ANUDEM method that estimated terrain elevation levels below sea level, giving a much more realistic DEM. The other methods, Natural Neighbor and TIN gridding, both estimated the terrain to be above sea level, but flatten out at sea level, since no contour data exists below sea level in our input data. Without considering those low elevation points, our automated methods built from contour points seemed to match the terrain almost identical to the ANUDEM method, with RMSE values around 17.45 m.

As mentioned, we wanted to also compare DEMs generated from only random elevation points instead of the contour points. The resulting RMSE values for those DEM were significantly higher, with values around 2867.71 m. The large discrepancy in RMSE values is likely due to the fact that the elevation points are not evenly distributed across the terrain, and thus do not provide a good representation of the true terrain.

5.1 Hillshade and slope

Hillshade and slope results were presented in Figures 1 and 2, respectively. These again confirm that the reccomended approach to DEM generation is to use contour points.

The middle columns of both figures shows the results of the elevation points, which are blurred, very coarse, and not really informative. This was expected, since we did zero preprocessing on the elevation points, and they are not evenly distributed across the terrain.

The stream burn approach seems to give the most fine-grained resolution, on both figures. Though, the automated methods and ANUDEM also appear very detailed and informative.

When zooming in further, it becomes apparent that the ANUDEM had the *least amount of artifacts*. Both of the simple automated methods had some obvious artifacts across the fjords, which were not present in the ANUDEM method. Though it should be mentioned that judging by the slope results alone, the natural neighbor method had very little artifacts, and seemed quite comparable to the ANUDEM method.

6 Conclusion

In this report, we have explored different methods for generating DEMs using contour lines and elevation points. We have also performed a quality assessment of the DEMs using RMSE, hillshade, slope, and profile analysis.

As much as we could, we automated this process in order to be able to quickly rerun it on a new geodatabase. Some limitations to the automation was the need to use ArcGIS Pro for the ANUDEM method, and the need to manually prepare the data as a geodatabase. Everythiing was possible to implement in Python, generating results that were similar to the ANUDEM method.

For the best results, we recommend using the ANUDEM method, as it produced the most realistic DEM, including the low elevation points not represented by our contour points. However, due to being proprietary software, the ANUDEM method is not as accessible as the other methods. Simpler methods like TIN gridding and natural neighbor interpolation are also good options, especially when working with rich data sets.

When possible, we recommend using points from contour lines rather than stand-alone elevation points to generate DEMs. Contour layers are often much more informative, as long as they have been generated from a realiable source. In our case, using the N50 data from Kartverket, with a FAIR status of 0.95, we can be confident that the contour lines are a good representation of the terrain for this region.

7 References

- John Lindsay et. al, *Whitebox tools*, `https://github.com/jblindsay/whitebox-tools`
- Sean Gillies et. al, *Rasterio*, `https://github.com/rasterio/rasterio`
- GRASS Development Team, *GRASS GIS*, https://grass.osgeo.org
- ESRI, *ArcGIS Pro*, `https://www.esri.com/en-us/arcgis/products/arcgis-pro/overview`
- Hutchinson, M.F., 1989. A new procedure for gridding elevation and stream line data with automatic removal of spurious pits. *Journal of Hydrology*, 106(3-4), pp.211-232. https://doi.org/10.1016/0022-1694(89)90073-5, (ANUDEM)
- Kartverket, *N50 Kartdata*, `https://kartkatalog.geonorge.no/metadata/n50-kartdata/ea192681-d039-42ec-b1bc-f3ce04c189ac?search=n50`