AGH

University of Science and Technology in Krakow

DEPARTMENT OF MINING GEODESY AND ENVIRONMENTAL ENGINEERING

Theme: “SAR, Interferometry - Project 2 - Induced seismicity”

Author: Bartosz Augustyn, Tymoteusz Maj

Field of study: Remote Sensing and GIS

Kraków, 2024

Introduction

Induced seismic activity, often resulting from mining operations, can lead to noticeable ground surface displacements. This report focuses on analyzing surface movements triggered by a mining-induced earthquake that occurred on November 29, 2016, at one of KGHM's mining sites. Using Differential Interferometric Synthetic Aperture Radar (DInSAR) techniques, the project investigates the extent of ground displacement and its progression over time, providing insights into the dynamics of the phenomenon.

Steps taken to perform the task

1. Downloading satellite data

Sentinel-1 radar images were downloaded from the Copernicus Data Space Ecosystem service by logging onto a dedicated platform. Two SLC (Single Look Complex) products with the following identifiers were selected:

- **S1B_IW_SLC__1SDV_20191208T163422_20191208T163449_019276_024651_E2F6**
- **S1B_IW_SLC__1SDV_20191220T163421_20191220T163448_019451_024BE6_79C7**

2. SNAPHU software installation

Installed the SNAPHU Unwrapping plug-in in the SNAP software to develop the interferometric phase. Additional tool packages were installed using the Manage External Tools function.

3. Data import and initial exploration

- Sentinel-1 files were imported into SNAP and visually inspected.

4. Co-registration of radar images

Co-registration of two radar images was carried out for accurate matching:

- **TOPS splitting:** IW2 sub-band and VV polarization splitting was performed using the S-1 TOPS Split operator.
- **Application of orbit information:** Precise orbit files were implemented using the Apply Orbit File operator.
- **Back Geocoding**: A co-registered stack was created using NMT data (SRTM 1Sec HGT).
- **Improved Spectral Diversity:** The Enhanced Spectral Diversity (ESD) operator was used to improve co-registration.

Co-registration results were visually verified, creating an RGB representation of the main and sub images.

5. interferogram generation and coherence estimation

- **The Interferogram Formation** operator was used to generate the interferogram and calculate coherence for the study area.
- The flat ground and topographic phase were eliminated.

6. data filtering and trimming

- **TOPS band merging:** Offsets between bands were removed using the TOPS Deburst operator.
- **Goldstein phase filtering**: Phase filtering was applied to reduce noise.
- **Clipping:** Separated the study area of interest using the Subset operator, entering the clipping coordinates.

7. Interferometric phase expansion

- Exported the filtered interferogram to a format acceptable to SNAPHU.
- Performed interferometric phase expansion in SNAPHU and imported the expanded data back into SNAP.

8. Phase to Displacement conversion

- The Phase to Displacement operator was used to convert the developed phase to displacement metric values.

9. Terrain Correction

- Terrain correction was performed using the Range Doppler Terrain Correction operator, using NMT SRTM 1Sec HGT and WGS84 projection.

10. Creating a coherence mask

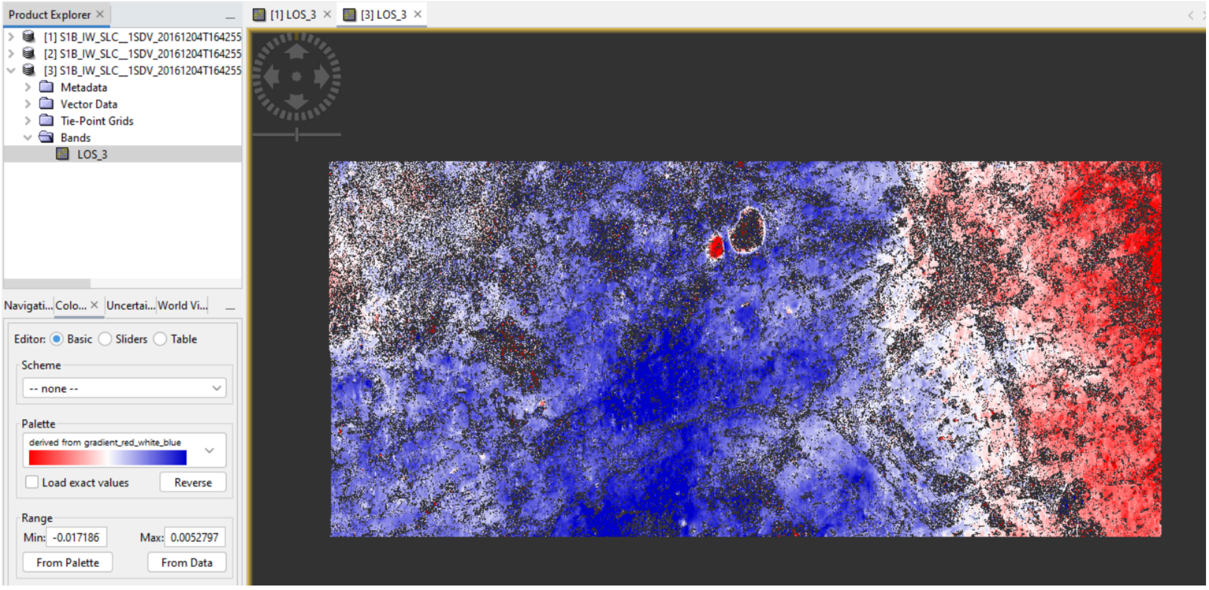
- A coherence mask was created, defining areas with coherence values greater than 0.3.

11. Calculating metric values for surface displacement in LOS

- Using the Raster Calculator, we calculated the metric values of surface displacement in the LOS direction by applying the previously created coherence mask

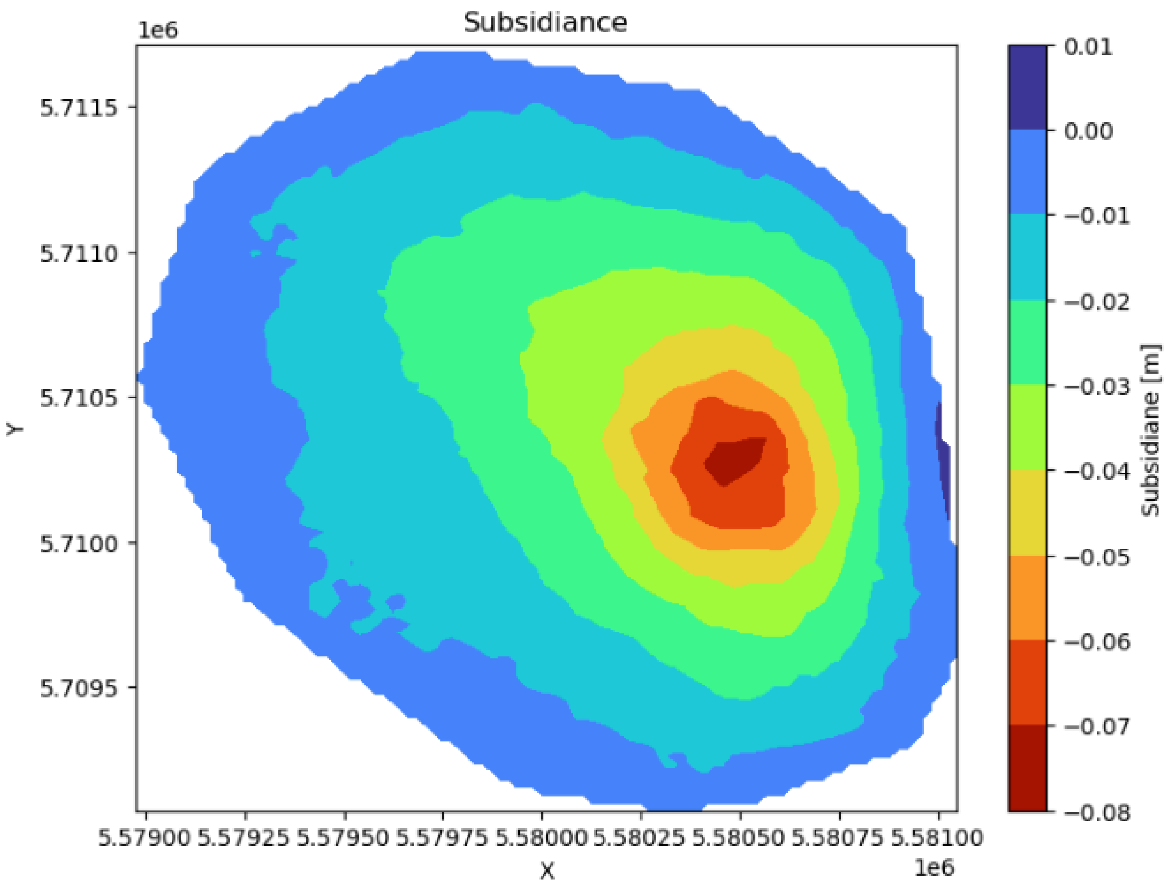
12. Applying multilooking

- To enhance the quality of the interferometric product and reduce noise, we applied the **multilooking** process.



13. Interpolation of results using a Python script

- To further refine the displacement analysis, we performed interpolation of the results using a custom **Python script**.

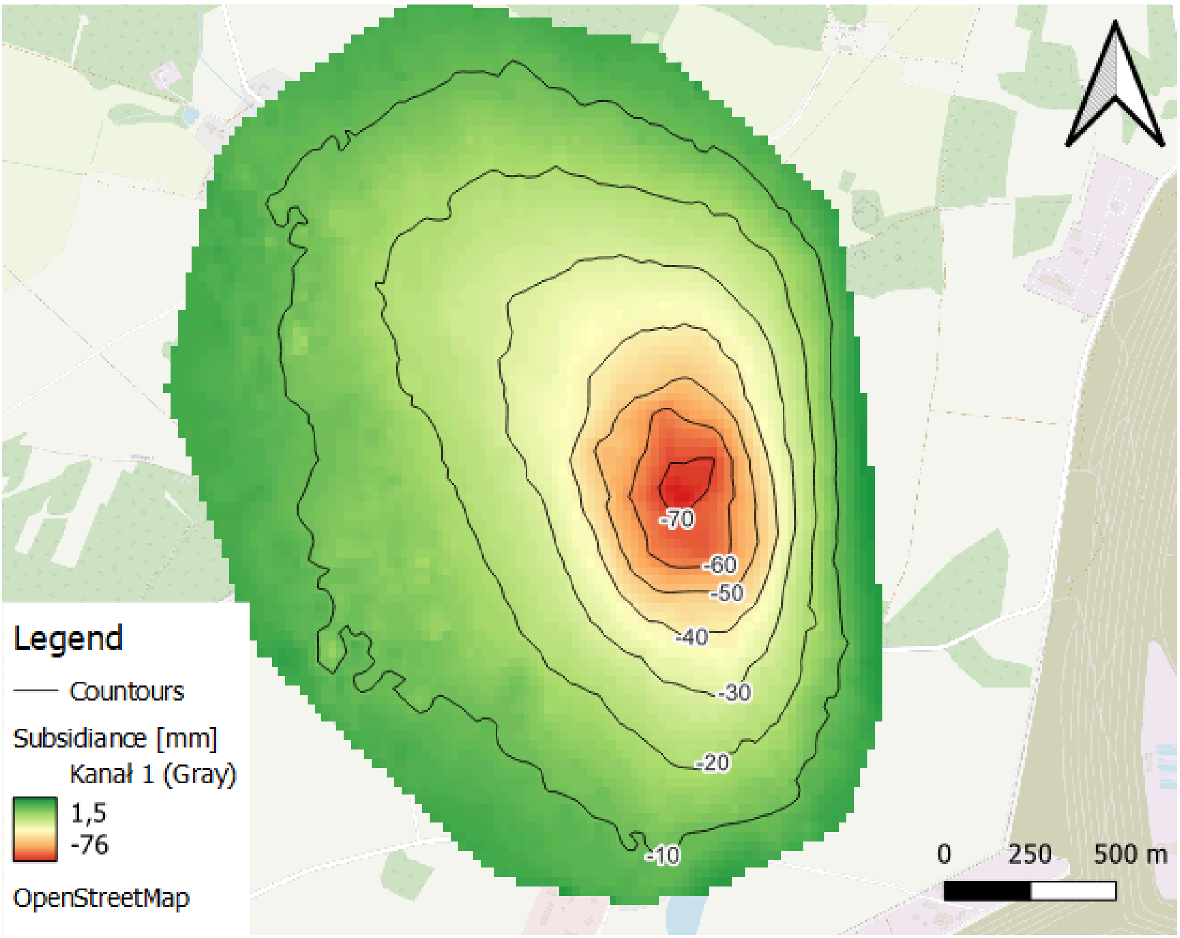


```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.interpolate import griddata
import rasterio
from rasterio.transform import from_origin

file_path = "D:\Pliki na AGH\IGP-RES\witkowski\SAR_Inferometry\Projekt2_Tremor\obliczenia\punkty.csv"
data = pd.read_csv(file_path)

points = np.column_stack(((data['left'] + data['right']) / 2, (data['top'] + data['bottom']) / 2))
z_values = data['SAMPLE_1'].values

xi = np.linspace(points[:, 0].min(), points[:, 0].max(), 100)
yi = np.linspace(points[:, 1].min(), points[:, 1].max(), 100)
grid_x, grid_y = np.meshgrid(xi, yi)

grid_z = griddata(points, z_values, (grid_x, grid_y), method='linear')
grid_z_flipped = np.flipud(grid_z)

output_tiff_path = "D:\Pliki na AGH\IGP-RES\witkowski\SAR_Inferometry\Projekt2_Tremor\subsidiance.tif"
transform = from_origin(xi[0], yi[-1], xi[1] - xi[0], yi[1] - yi[0])

with rasterio.open(
    output_tiff_path,
    "w",
    driver="GTiff",
    height=grid_z.shape[0],
    width=grid_z.shape[1],
    count=1,
    dtype=grid_z.dtype,
    crs="EPSG:2176",
    transform=transform,
) as dst:
    dst.write(grid_z_flipped, 1)

plt.figure(figsize=(8, 6))
plt.contourf(grid_x, grid_y, grid_z, levels=7, cmap='turbo_r')
plt.colorbar(label='Subsidiane [m]')
plt.title('Subsidiance')
plt.xlabel('X')
plt.ylabel('Y')
plt.show()
```

14. Visualization of results in QGIS and assigning appropriate contour lines

- To effectively present the displacement analysis, we visualized the results in **QGIS**, leveraging its advanced cartographic tools. The processed raster data was imported into the project, where appropriate symbology was applied to highlight areas of significant displacement.
- Using QGIS's **Contour tool**, we generated contour lines to represent displacement values at defined intervals. These lines provided a clear visual representation of displacement gradients and facilitated the interpretation of spatial patterns.

Results



-70
-60
-50
-40
-30
-20
-10
Legend
Countours
Subsidiance [mm]
Kanał 1 (Gray)
1,5
-76
OpenStreetMap
0
250
500 m