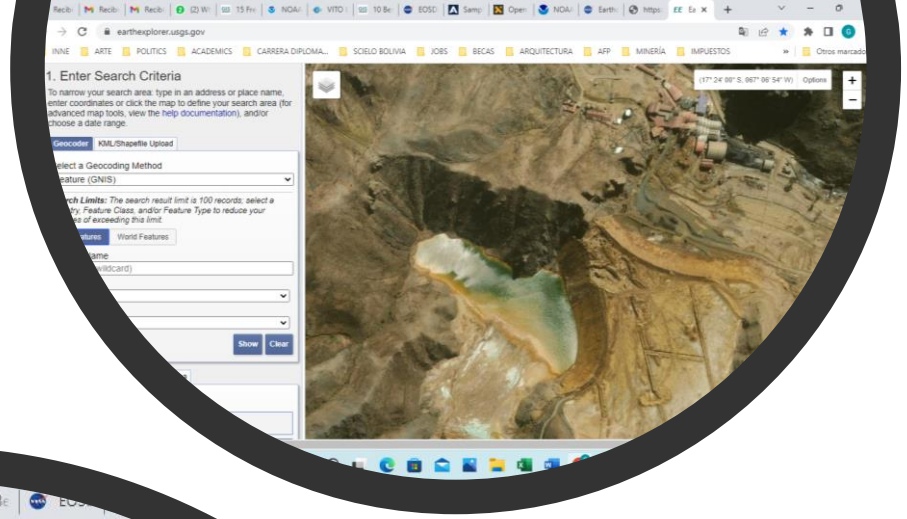
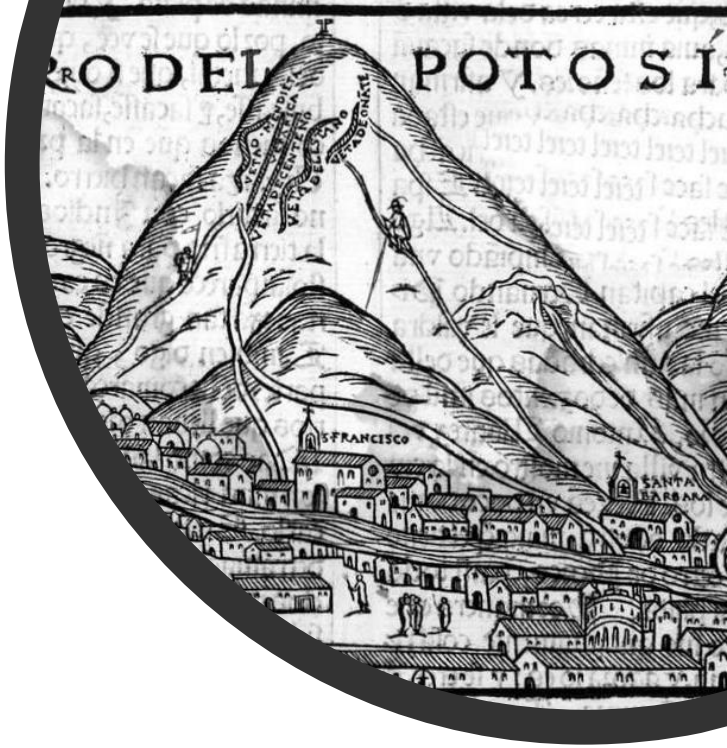
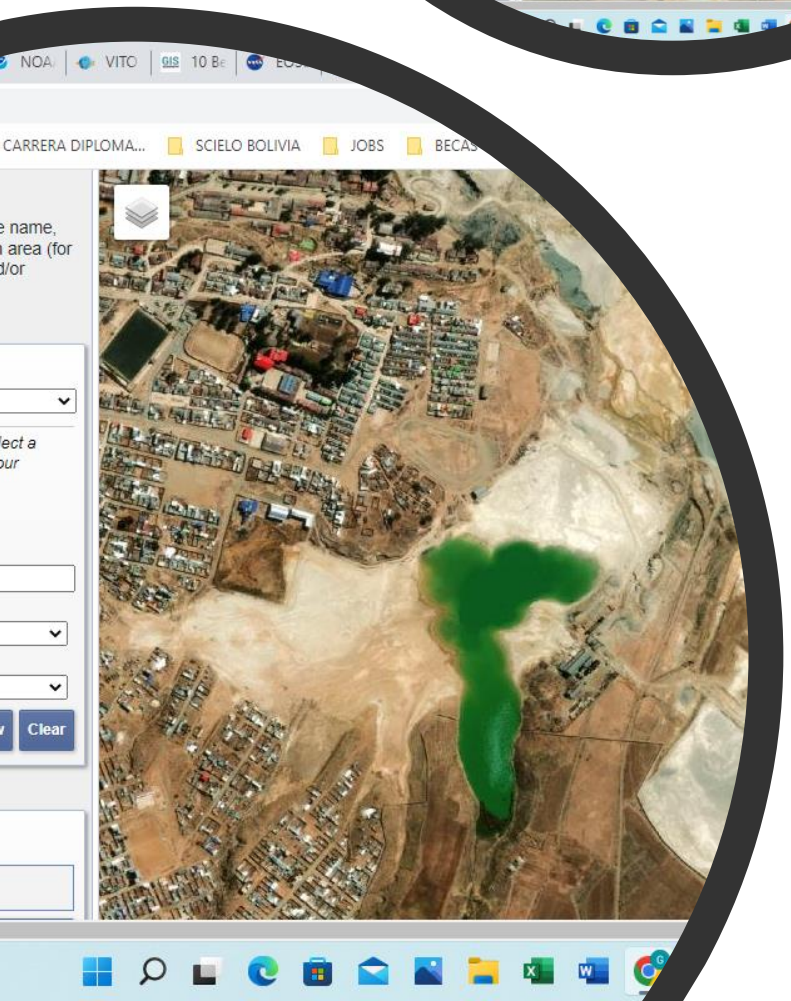
Identificación satelital de pasivos ambientales mineros en Bolivia ISPAM

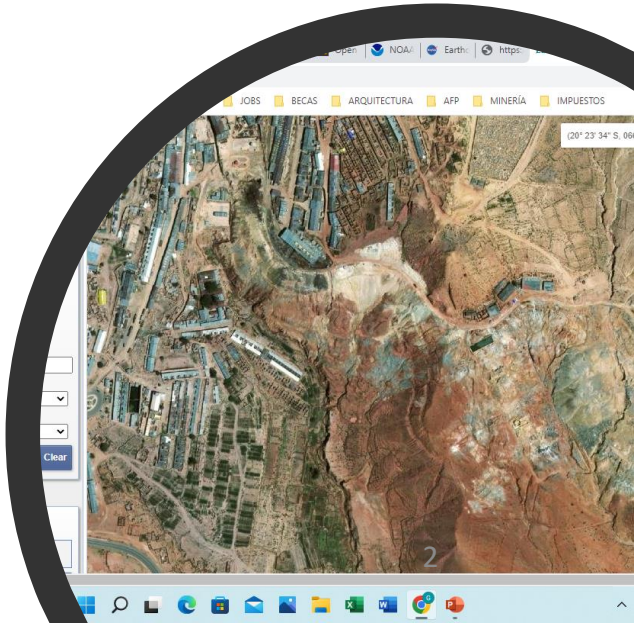






Problema: 1545

ISPAM

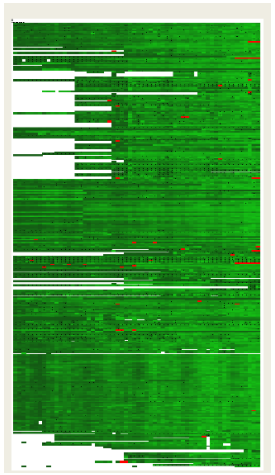


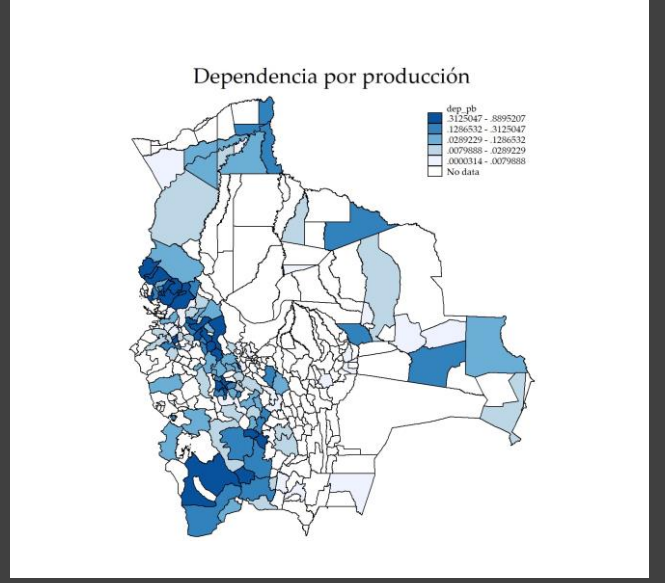
ISPAM

- Dr. Guillermo Guzmán Prudencio

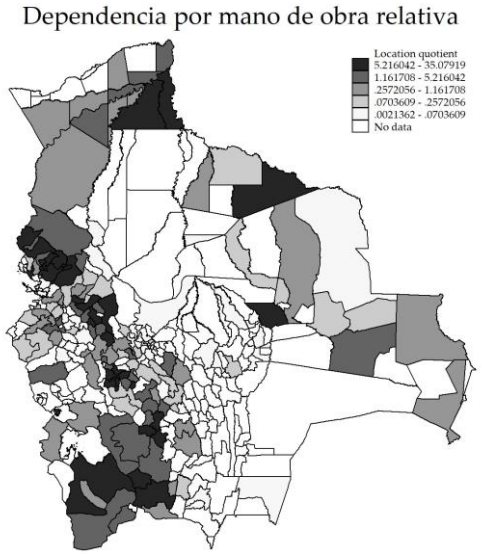
- Dr. Ariel Zeballos

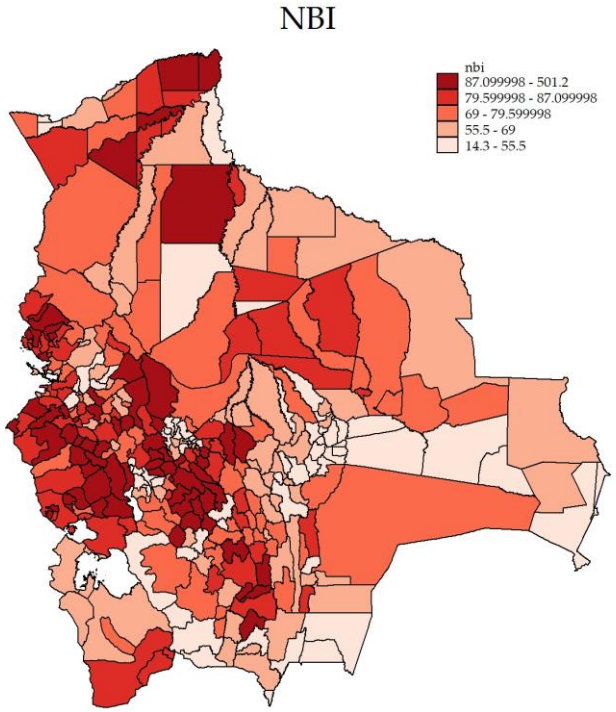
- Ing. Adrián Villarroel

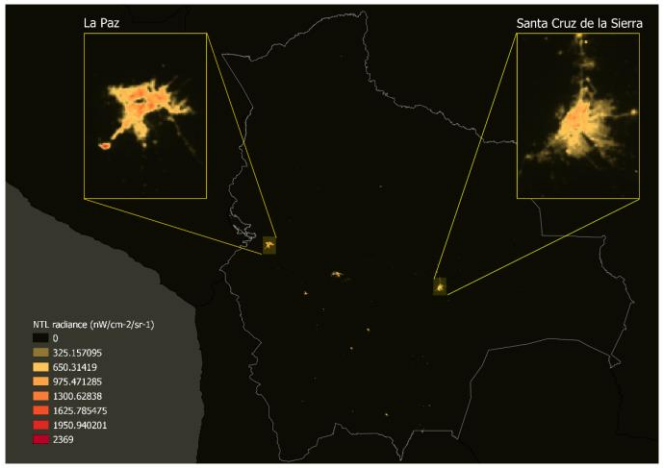



Dependencia por producción
dep_pb
.3125047 - .8895207
.1286532 - .3125047
.0299229 - .1286532
.0079888 - .0299229
.0000314 - .0079888
No data

ISPAM


Dependencia por mano de obra relativa
Location quotient
5.216042 - 35.07919
1.161708 - 5.216042
.2572056 - 1.161708
.0703609 - .2572056
.0021362 - .0703609
No data


NBI
nbi
87.099998 - 501.2
79.599998 - 87.099998
69 - 79.599998
55.5 - 69
14.3 - 55.5


La Paz
Santa Cruz de la Sierra
NTL radiance (nW/cm-2/sr-1)
0
325.157095
650.31419
975.471285
1300.62838
1625.785475
1950.940201
2369

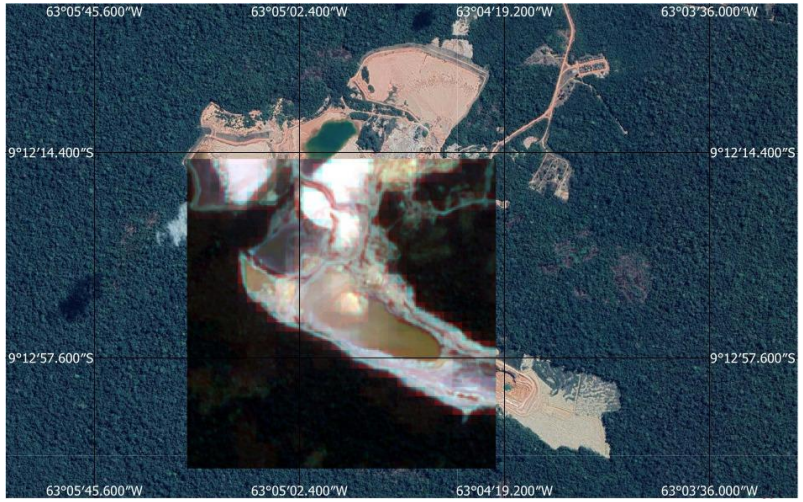
Research: revisión bibliográfica ISPAM

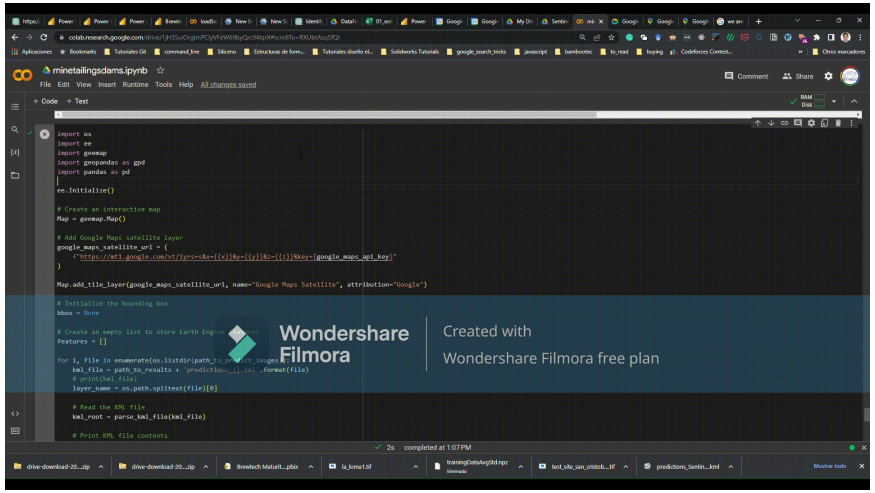
- Armstrong, M., Petter, R., Petter, C., 2019. Why have so many tailings dams failed in recent years? Resources Policy.
- Carlà, T., Intrieri, E., Raspini, F., Bardi, F., Farina, P., Ferretti, A., Colombo, D., Novali, F., Casagli, N., 2019. Perspectives on the prediction of catastrophic slope failures from satellite InSAR. Scientific Reports.
- Costa, C., 2019. Brumadinho: Brasil tem mais de 300 barragens de mineração que ainda não foram fiscalizadas e 200 com alto potencial de estrago. https://www.bbc.com/portuguese/brasil-47056259.
- DNPM, 2019. Agência nacional de mineração: Cadastro nacional de barragens de mineração. http://www.anm.gov.br/assuntos/barragens/pasta-cadastro-nacional-de-barragens-de-mineracao.
- Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., Moore, R., 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sensing of Environment.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition.
- Huang, G., Liu, Z., Van Der Maaten, L.,Weinberger, K. Q., 2017. Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition.
- Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size. arXiv preprint arXiv:1602.07360.
- IBRAM, 2017. Information and Analyses on the Brazilian Mineral economy. Brazilian Mining Association – Instituto Brasileiro de Mineração (IBRAM). http://ibram.org.br/sites/1400/1457/00000380.pdf.
- ICOLD, U., 2001. Tailings Dams–Risk of Dangerous Occurrences, Lessons Learnt From Practical Experiences (Bulletin 121). Commission Internationale des Grands Barrages, Paris.
- Krizhevsky, A., 2014. One weird trick for parallelizing convolutional neural networks. arXiv preprint arXiv:1404.5997.
- Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., Johnson, B. A., 2019. Deep learning in remote sensing applications: A metaanalysis and review. ISPRS Journal of Photogrammetry and Remote Sensing.
- Masek, J. G., 2019. Landsat Science: Landsat 8 Overview. https://landsat.gsfc.nasa.gov/landsat-8/landsat-8-overview/.
- Morgenstern, N. R., 2001. Geotechnics and mine waste management–update. Seminar on Safe Tailings Dam Constructions.
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., De- Vito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A., 2017. Automatic differentiation in PyTorch. NIPS Autodiff Workshop.
- Simonyan, K., Zisserman, A., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision. Proceedings of the IEEE conference on computer vision and pattern recognition.

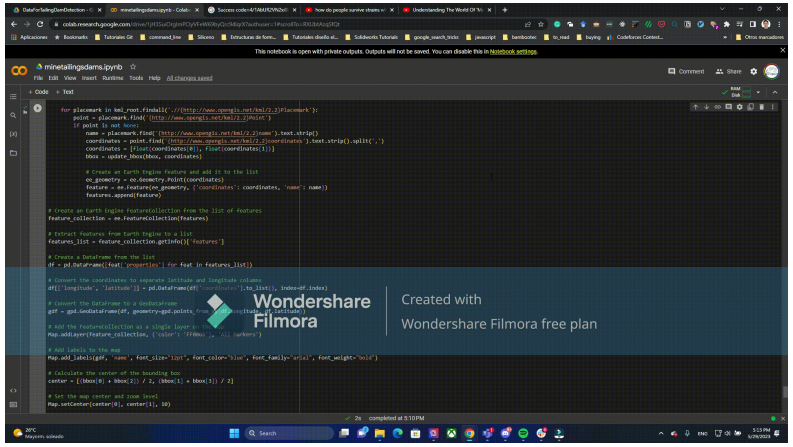
El algoritmo: ***Visual Pattern Mining*** (VPM)

IA-ML

- **Fase 1 (Morfológica)**: el algoritmo recibe el parche que indica que un subconjunto de esa imagen es el objeto que se busca, entonces el algoritmo clasifica partes del parche como pertenecientes al objeto de interés o no.
- **Fase 2 (Macro morfológica)**: el algoritmo establece relaciones espaciales entre estas partes clasificadas para formar el objeto de interés y clasificarlo correctamente en su totalidad.
- **Fase 3 (Espectrométrica):** Para evaluar el impacto, se tiene un segundo modelo que sólo indica si la espectrometría corresponde a un elemento que tiene alto, medio o bajo impacto. (12 bandas espectrales)




minetailingsdams.ipynb
Wondershare Filmora
Created with
Wondershare Filmora free plan


minetailingsdams.ipynb
Wondershare Filmora
Created with
Wondershare Filmora free plan

minetailingsdams.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
import os
import ee
import geemap
import geopandas as gpd
import pandas as pd

ee.Initialize()

# Create an interactive map
Map = geemap.Map()

# Add Google Maps satellite layer
google_maps_satellite_url = (
    f"https://mt1.google.com/vt/lyrs=s&x={{x}}&y={{y}}&z={{z}}&key={google_maps_api_key}"
)

Map.add_tile_layer(google_maps_satellite_url, name="Google Maps Satellite", attribution="Google")

# Initialize the bounding box
bbox = None

# Create an empty list to store Earth Engine features
features = []

for i, file in enumerate(os.listdir(path_to_predict_images)):
    kml_file = path_to_results + 'predictions_{}.kml'.format(file)
    # print(kml_file)
    layer_name = os.path.splitext(file)[0]

    # Read the KML file
    kml_root = parse_kml_file(kml_file)

    # Print KML file contents
```

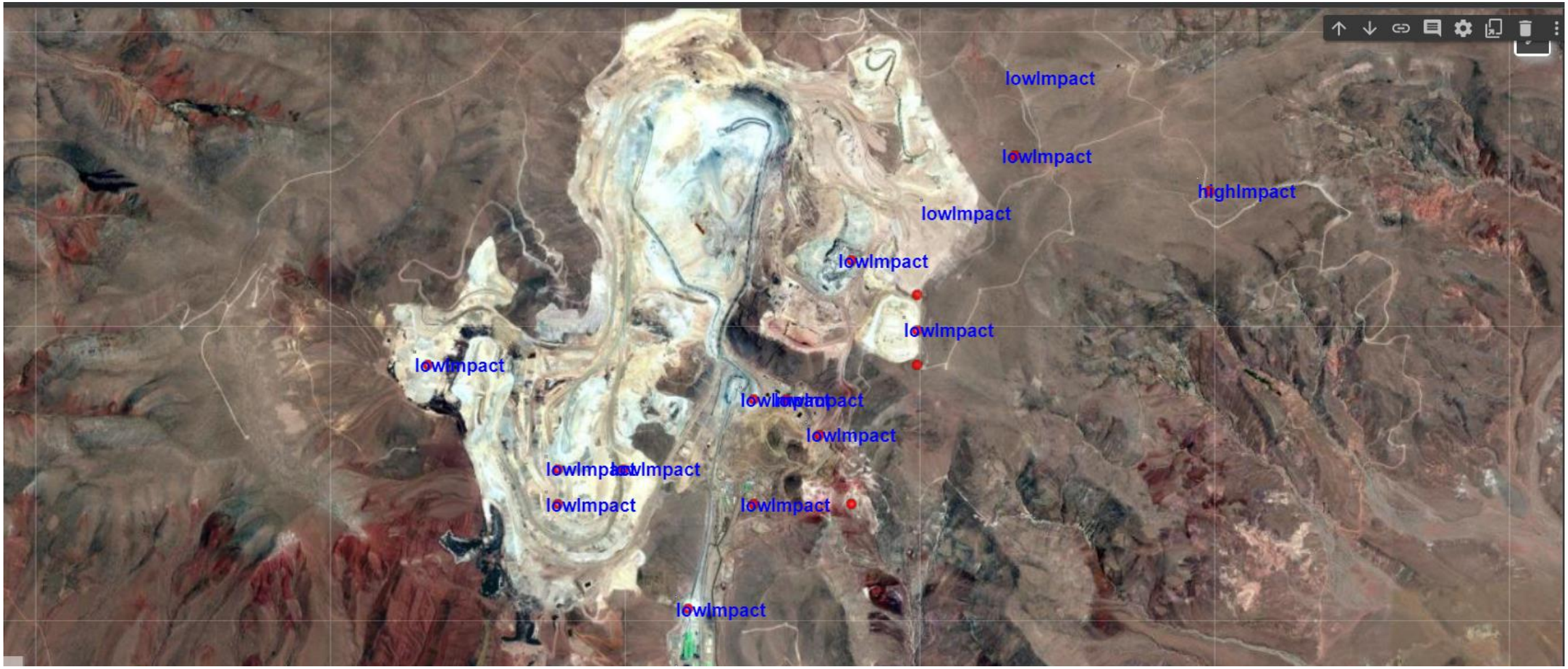

2s completed at 1:07 PM

ISPAM

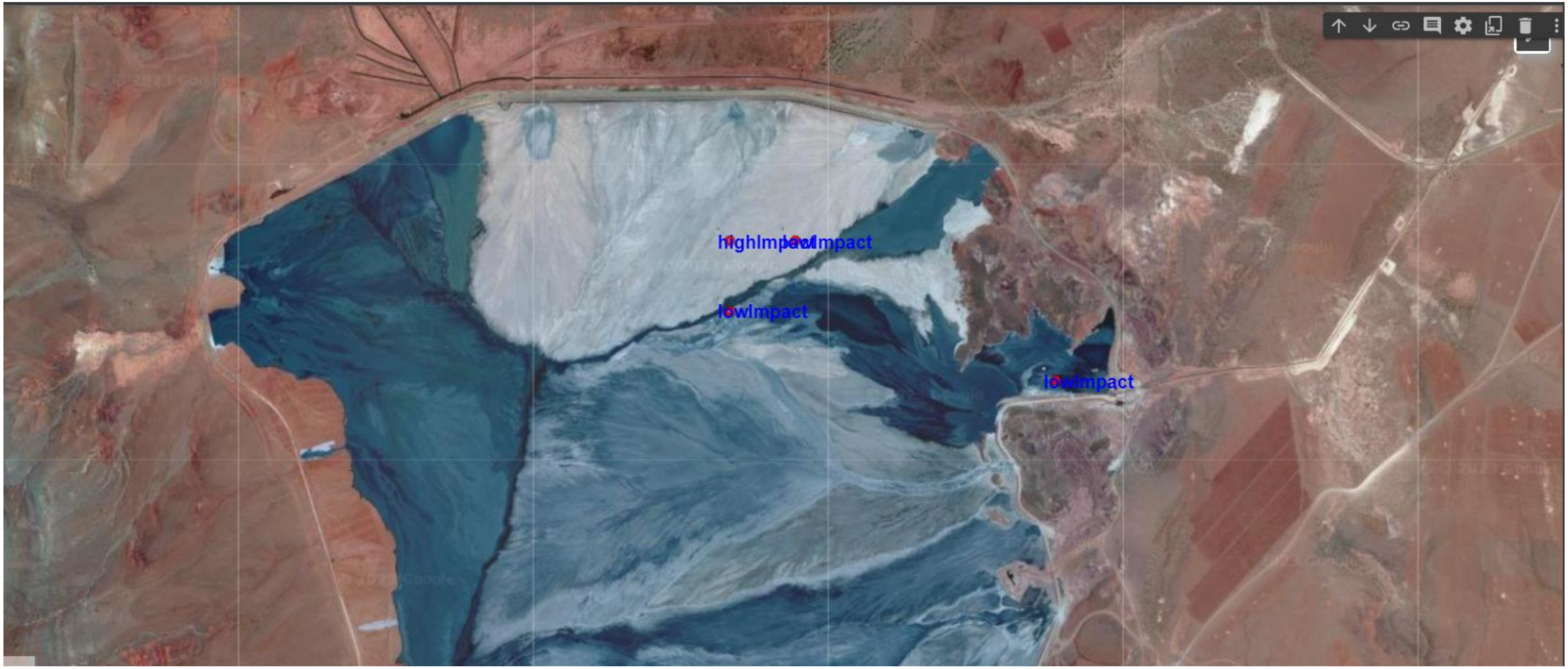
ISPAM


lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact

ISPAM


lowImpact
lowImpact
highImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact
lowImpact

ISPAM



DataForTailingDamDetection - G | minetailingsdams.ipynb - Colabo | Success code=4/1AbUR2VN2xi0 | how do people survive strains wh | Understanding The World Of 'Ma

colab.research.google.com/drive/1jH3SuiOrgImPCIyVFeW69byQrc94IqrX?authuser=1#scrollTo=RXUbtAzqSfQt

Aplicaciones | Bookmarks | Tutoriales Git | command_line | Siliceno | Estructuras de form... | Tutoriales diseño el... | Solidworks Tutorials | google_search_tricks | javascript | bambootec | to_read | buying | Codeforces Contest... | Otros marcadores

This notebook is open with private outputs. Outputs will not be saved. You can disable this in Notebook settings.

minetailingsdams.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

```
    for placemark in kml_root.findall('.//{http://www.opengis.net/kml/2.2}Placemark'):
        point = placemark.find('{http://www.opengis.net/kml/2.2}Point')
        if point is not None:
            name = placemark.find('{http://www.opengis.net/kml/2.2}name').text.strip()
            coordinates = point.find('{http://www.opengis.net/kml/2.2}coordinates').text.strip().split(',')
            coordinates = [float(coordinates[0]), float(coordinates[1])]
            bbox = update_bbox(bbox, coordinates)

            # Create an Earth Engine feature and add it to the list
            ee_geometry = ee.Geometry.Point(coordinates)
            feature = ee.Feature(ee_geometry, {'coordinates': coordinates, 'name': name})
            features.append(feature)

# Create an Earth Engine FeatureCollection from the list of features
feature_collection = ee.FeatureCollection(features)

# Extract features from Earth Engine to a list
features_list = feature_collection.getInfo()['features']

# Create a DataFrame from the list
df = pd.DataFrame([feat['properties'] for feat in features_list])

# Convert the coordinates to separate latitude and longitude columns
df[['longitude', 'latitude']] = pd.DataFrame(df['coordinates'].to_list(), index=df.index)

# Convert the DataFrame to a GeoDataFrame
gdf = gpd.GeoDataFrame(df, geometry=gpd.points_from_xy(df.longitude, df.latitude))

# Add the FeatureCollection as a single layer on the map
Map.addLayer(feature_collection, {'color': 'FF0000'}, 'All Markers')

# Add labels to the map
Map.add_labels(gdf, 'name', font_size="12pt", font_color="blue", font_family="arial", font_weight="bold")

# Calculate the center of the bounding box
center = [(bbox[0] + bbox[2]) / 2, (bbox[1] + bbox[3]) / 2]

# Set the map center and zoom level
Map.setCenter(center[0], center[1], 10)
```


2s completed at 5:10 PM

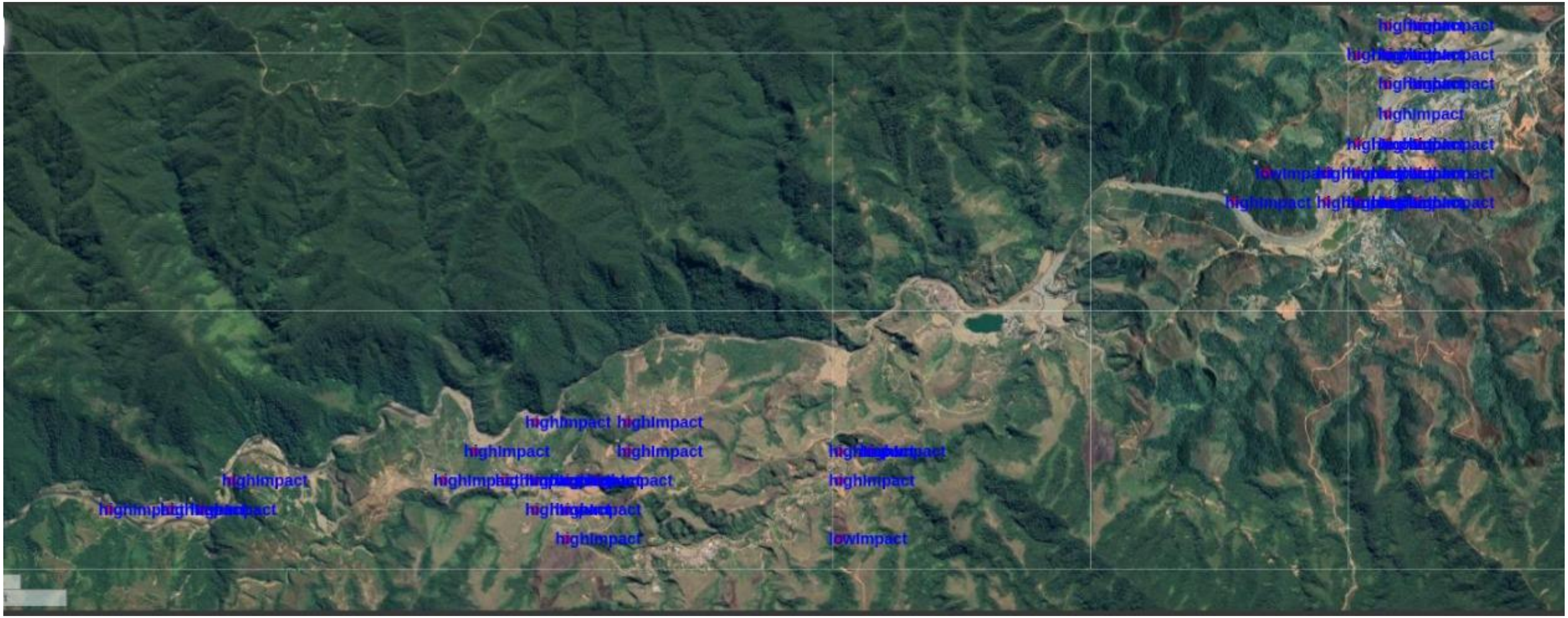
26°C Mayorm. soleado

Search

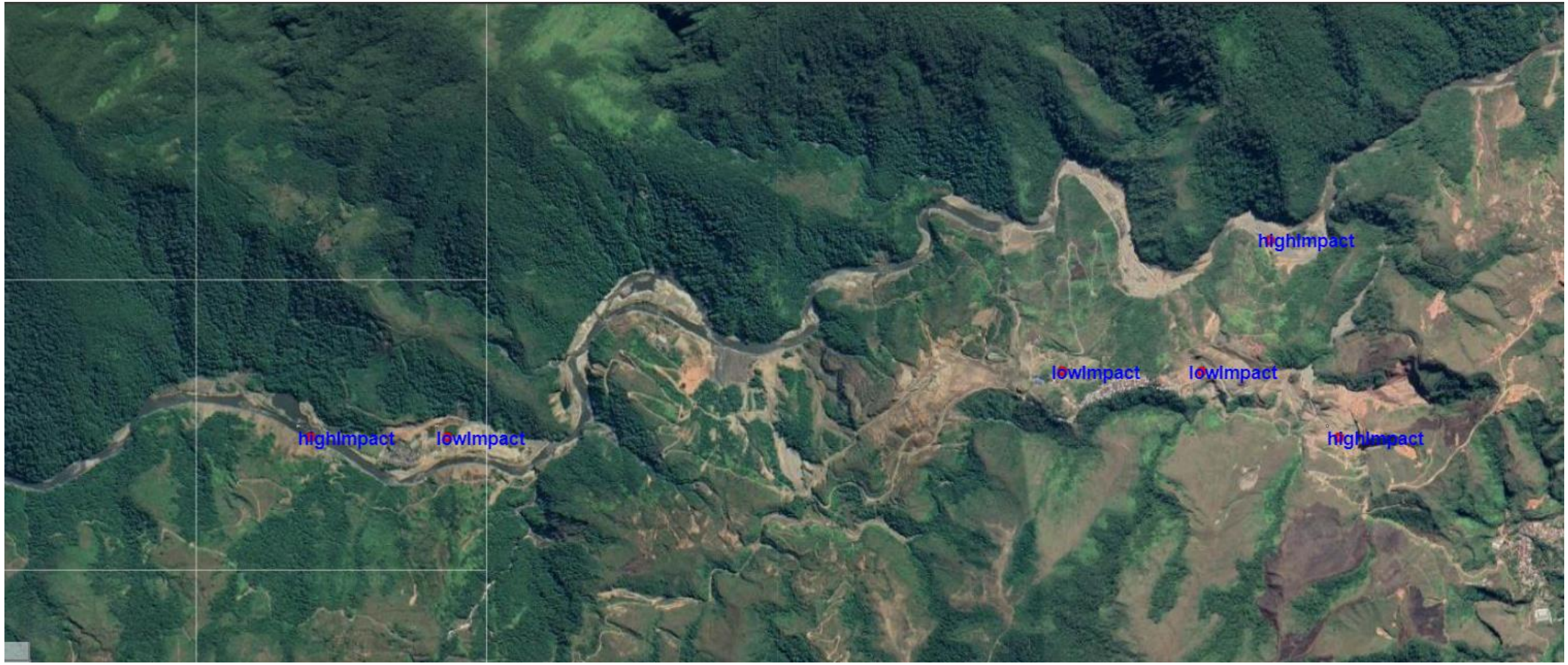
ENG 5:13 PM 5/29/2023

ISPAM

ISPAM


highimpact
highimpact highimpact
highimpact
highimpact
highimpact
highimpact
highimpact
highimpact
lowimpact

ISPAM


highImpact
lowImpact
highImpact
lowImpact
lowImpact
highImpact

ISPAM


highimpact
highimpact
highimpact
highimpact
highimpact
highimpact
highimpact
highimpact

Muchas gracias

- Dr. Guillermo Guzmán Prudencio
g_guzman_p@usal.es

- Dr. Ariel Zeballos
ariel.zeballos@bluewin.ch

- Ing. Adrián Villarroel
adrianvn789@gmail.com

ISPAM