GIS Analysis of debris volumes following recent hostilities in the Gaza Strip:

As this is a volunteer initiative and not everyone has access to ArcGIS Pro, this demo utilises QGIS open-source GIS software. Arc Pro will be used in cases when QGIS is not appropriate such as the creation of ESRI StoryMaps.

This analysis is based on the premise that the total estimated weight of the building (in metric tons) can be calculated as:
Wtotal =A×D× N×C

Where:

- **Wtotal** = Total estimated weight of the building (in metric tons)
- **A** = Building footprint (in m^2)
- **D** = Level of destruction as a percentage
- **N** = Number of floors
- **C** = A combined average weight factor (in metric tons/m^2/floor) – In this case initially 2 which is to be confirmed by the project team.

The building footprints are from Humanitarian OpenStreetMap available on HDX. We can use this footprint layer as our main layer and add attributes for level of damage (from UNOSAT) and building height (from the GEDI study) to derive debris volumes per OSM building.

1) Preparing QGIS project

a) Create a new project

i. Concerning CRS: The files sourced don't all use the same Coordinate Reference System

Layer	CRS	CRS Type
Admin levels	EPSG:4326 - WGS 84	geographic
Building Footprints	EPSG:4326 - WGS 84	geographic
Destruction levels	EPSG:4326 - WGS 84	geographic
Building Heights	WGS_1984_Mollweide	projected

It is best to use a common CRS across the project and a Projected Coordinate System (PCS) may be preferable due to higher accuracy when measuring distances and areas, a PCS also can be expressed in units of meters rather than degrees as with a Geographic Coordinate System (GCS).

For the purpose of this demo Palestine Grid (EPSG:28191) will be used as the common CRS, the final choice should be considered by the whole project team.

ii. Select basemap

iii. Import the geo-files to be used

- Admin levels: HDX State of Palestine - Subnational Administrative Boundaries
- Building Footprints: HDX State of Palestine Buildings (OpenStreetMap Export)
- Destruction levels: UNOSAT Gaza Strip Comprehensive Damage Assessment
- Building Heights: GEDI 150-m Global Urban Building Height Data (circa 2020)

b) Establish the geographic extent of the analysis (The Gaza Strip). Building height estimations are global datasets and the OSM footprints cover all of Palestine not just the Gaza Strip, including only the required data will provide clarity and improve speed of processing.

i. Filter administrative levels 1 and 2 for The Gaza Strip only and excluding the West Bank

- Admin 1 will be used as a mask to define extent of analysis
- Admin 2 may be used for disaggregation analysis if desired

ii. Use Gaza Strip admin1 as a mask to filter Building Footprints and Building Heights.

- Add spatial indexes first to improve processing speeds.
- Building Footprints: Menu items-> Vector/Geoprocessing Tools/Clip
- Building Heights: Menu items-> Raster/Extraction/Clip Raster by Mask Layer

iii. Reproject Layers to common CRS. Reprojecting after clipping to extent greatly speeds up the process especially for the building heights raster file.

2) **Quantify level of damage:** We need to quantify the damage estimates provided in the UNOSAT layer's "Main_Damage_Site_Class_##" attribute. This involves converting their categorical values into numerical ones so we can calculate estimated debris volumes. (* Estimated % destroyed needs to be discussed and agreed)

Value	Mapped Text	* Est % Destroyed
1	Destroyed	100
2	Severe Damage	75
3	Moderate Damage	50
4	Possible Damage	25
6	No Visible Damage	0
5	Impact Crater (Damage to Road)	0
7	Impact Crater (Damage to Field)	0
8	Destroyed (Demolition Zone)	100
9	Suspected Phosphorus shell Attack Zone	0
10	Road Damage from Tank Passage	0
11	Possible Damage from adjacent impact, debris	15
12	Lava Flow	0
13	Damage	0

a) Create a new attribute field “Damage_PCT” in the UNOSAT layer using the field_calculator with a formula such as:

```
CASE
  WHEN "Main_Damage_Site_Class_12" IN(1,8) THEN 100
  WHEN "Main_Damage_Site_Class_12" = 2 THEN 75
  WHEN "Main_Damage_Site_Class_12" = 3 THEN 50
  WHEN "Main_Damage_Site_Class_12" = 4 THEN 25
  WHEN "Main_Damage_Site_Class_12" = 6 THEN 0
  WHEN "Main_Damage_Site_Class_12" IN(5,7,9,10,12,13) THEN 0
  ELSE NULL -- For any unexpected values
END
```

There are other value map attributes in the UNOSAT layer that may be of interest to the project and should be considered for value in further analysis:

ConfedenceID_##	
1	Very High
2	Medium
3	Uncertain
4	To Be Evaluated

Damage_Status_##	
0	No change
1	Increase - damage
2	Decrease - damage
3	New - damage
4	Demolished
5	Rebuilt
6	Reconstructed

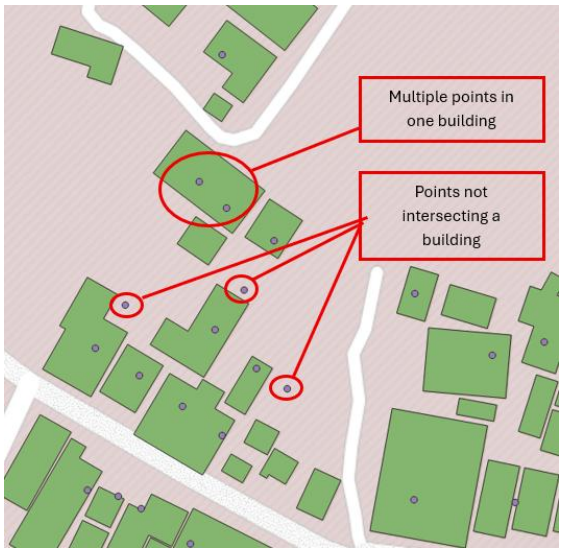
Grouped_Damage_Calsses	
1	Damaged Buildings
2	Damaged Greenhouses
3	Impacts on Roads / Bridges
4	Impacts in Fields
5	Damaged Road segments
6	Damaged Port

FieldValidation	
0	Not yet field validated
1	Validated (correct)
2	Validated (not correct)
3	Site Not Identified in Image

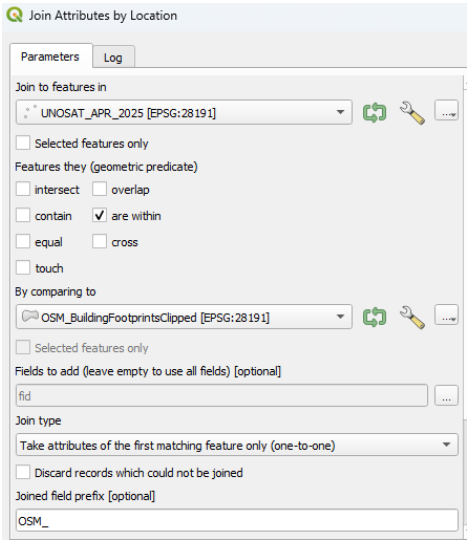
3) Add destruction values to the footprint layer: Now that we have quantitative destruction levels, we can associate them with building footprints.

Add destruction values to building footprint layer. At times the UNOSAT points do not align with OSM polygons. We need to associate each UNOSAT point to its closest OSM Building footprint polygon and then calculate the average damage value of the point values (if more than one per building) that intersect or are near to each polygon.

This can be done in two steps, first by working with points that are inside polygons then by assigning points outside of polygons to the nearest polygon (Working with just one step of nearest polygons may misassign points to neighbouring buildings if they are closer to the adjacent polygon’s centroid so the two-step approach can be used to avoid this).



a) Joining UNOSAT points with OMS polygons
   i. Points contained within polygons: Use a spatial join



This will produce a new Points layer with added attributes for the building footprint fid, attribute values will be null for points outside of polygons.

Duplicate the layer and delete all nulls in one copy and all values in the other to get a layer “points_inside” and a second “points_outside”.

Use the "Distance to nearest hub (line to hub)" tool to associate nearest building footprints with points outside of footprint polygons and use “Join attributes by field value” to import the building footprint id to the “points_outside” layer.

Merger the “points_inside” and “points_outside” layers to get a new UNOSAT layer copy with the added attribute of associated building id.

Run the "Statistics by categories" tool to get the mean value of Damage_PCT for each building footprint polygon. It may be decided later if the max value would be more appropriate.

Use the Join attributes by field value tool to add the aggregate mean Damage_PCT value as an attribute in the building footprint layer.

It is best to use a common CRS across the project and a Projected Coordinate System (PCS) may be preferable due to higher accuracy when measuring distances and areas, a PCS also can be expressed in units of meters rather than degrees as with a Geographic Coordinate System (GCS).

4) Add building height values to the footprint layer:

a) Run Zonal Statistics to assign estimated height values from GEDI raster file to the OSM Building footprints
   i. Use the “Zonal statistics” tool to attribute building height values to building footprint polygons.
   ii. Then Use the “Join on attribute value” to import building height values to the OSM building footprint layer

b) Use the Field calculator to calculate the number of floors to one decimal place, null values are to be considered 1 floor high using this SQL expression:

```
round(coalesce("Height_mean" / 3, 1), 1)
```

5) Calculate weight per building:

a) Use the Field calculator to calculate a new field for building footprint area

```
$area
```

b) Calculate the level of debris for each building using the formula **Wtotal =A×D×N×C**
Where:
- **Wtotal** = Total estimated weight of the building (in metric tons)
- **A** = Building footprint (in m^2) – Area calculated above
- **D** = Level of destruction as a percentage – UNOSAT mean level of destruction
- **N** = Number of floors – estimated number of floors
- **C** = A combined average weight factor assumed to be two

Using the field calculator (field names may differ):

```
"Area"  *  "UNOSAT_Dammage_pctmean"  *  "Est_Floors"  * 2
```

c) Optionally add symbology to colour the building footprints based on level of damage 0 green to 100 red with null uncoloured.

6) Other considerations:

- Taller buildings usually have larger footprints, calculation of height can be adjusted to incorporate this
- Add a Python script to count the selected buildings and provide a sum of their estimated debris weight similar to:

```
from qgis.utils import iface
from PyQt5.QtWidgets import QMessageBox

layer = iface.activeLayer()
if layer is not None and layer.selectedFeatureCount() > 0:
    selected = layer.selectedFeatures()

    damage_attr = "Mean_Damage_%"
    weight_attr = "Weight_tons"

    damage_values = []
    weight_total = 0

    for f in selected:
        # Handle Mean_Damage_%
        damage_val = f[damage_attr]
        try:
            damage_val = float(damage_val)
            damage_values.append(damage_val)
        except (TypeError, ValueError):
            pass  # skip invalid

        # Handle Weight_tons
        weight_val = f[weight_attr]
        try:
            weight_val = float(weight_val)
            weight_total += weight_val
        except (TypeError, ValueError):
            pass  # skip invalid

    valid_damage_count = len(damage_values)
    avg_damage = sum(damage_values) / valid_damage_count if valid_damage_count > 0 else 0

    # Format total weight: thousands separator, no decimals
    weight_formatted = f"{int(round(weight_total)):,}"

    QMessageBox.information(
        iface.mainWindow(),
        "Selection Summary",
        f"No damaged buildings: {valid_damage_count}\n"
        f"Ave % damage: {avg_damage:.2f}\n"
        f"Total est weight of debris: {weight_formatted} tons"
    )
else:
    QMessageBox.warning(iface.mainWindow(), "No Selection", "No features selected.")
```

The script can be used to create a tool that returns statistics for a selection of buildings: