



ai4ge

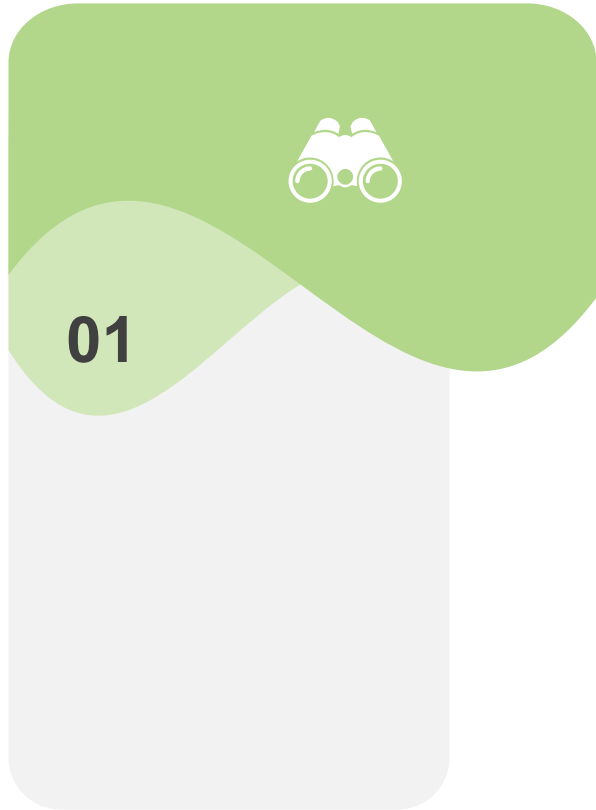
An automatic 3D geospatial information capability

Simon BAILLARIN (CNES)

simon.baillarin@cnes.fr

Outline

- ▶ **AI4GEO CONTEXT AND INITIATIVE**
- ▶ **AI4GEO DEVELOPMENT PLATFORM**
- ▶ **FIRST RESULTS**
- ▶ **PERSPECTIVES**



AI4GEO CONTEXT AND INITIATIVE

Geospatial information market booming, driven by demand for 3D



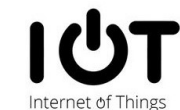
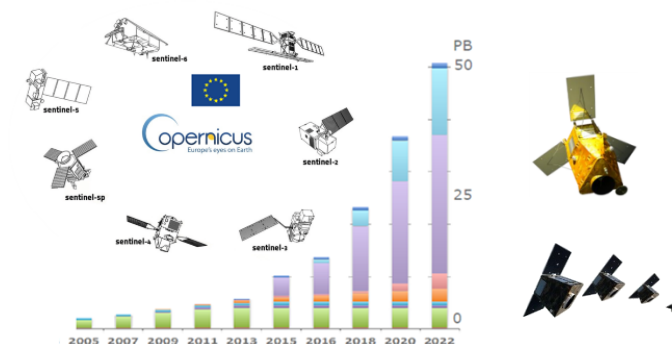
Annual growth > 12% **~\$100 billions in 2020**

Multiple markets in demand (urban monitoring, environment, economic intel, autonomous vehicles, ...)

At the same time, multiple sources of data abound, which multiplies opportunities

Accessible, global, high resolution (spatial, spectral, temporal)

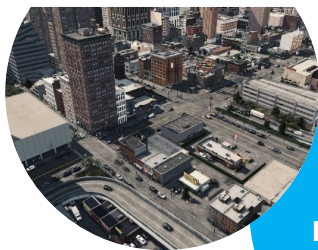
Bloomberg
Business
Geospatial Analytics Market Size Worth \$134.48 Billion by 2025: Grand View



Manufacturing costs and production times are too high

- AI4GEO project : to develop an automated solution for producing 3D geospatial information (3D objects)
 - Guarantee an homogeneous level of performance in reconstructing 3D surfaces
 - Process large coverage by distributing algorithms on a parallel infrastructure
 - Qualify the results according to standardized accuracy classes for downstream applications
- R&D project over 4 years, 2 axes in parallel, 9 partners.
 - AXIS 1: Technological building blocks focused on 3D imaging, segmentation and bulk processing.
Collaborative work
 - AXIS 2: Applicative components addressing 6 industrial sectors (urban mapping, environment, autonomous vehicles, ...).
Work carried out by each industrial partners.

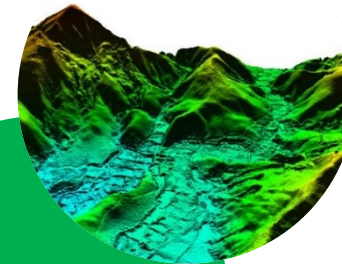
AXIS 2 : Applicative components



Urban monitoring

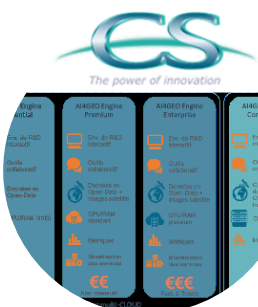
3D Urban Map:
Semantic 3D mapping of cities

AIRBUS



Environment

Decision support mapping
(Land use & water management)



Geospatial data management

GeoData platform



Customer market



QwantMap 3D:
3D Search Engine

Autonomous vehicles



Support for HD map production

Economic Intelligence



Urban growth indicator



AXIS 1 : a 4 years collaborative work plan

AI4GEO

T0+48

Urban : **LOD3** VHR with semantic
 Global : Classification in VHR, based on **30 classes**, **change detection** and **characterisation of activities**.
 Tests sites : 10 cities + 5 regions + 1 country
 AI4GEO Platform final version

T0+36

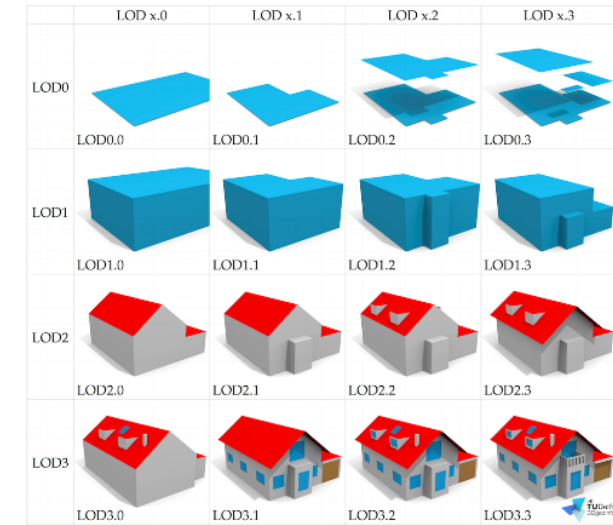
Urban : **LOD2** VHR (**0.5m**).
 Global : Classification in VHR, based on **20 classes** and **change detection**
 Tests sites : 10 cities + 5 regions
 Dev Platform 3rd version

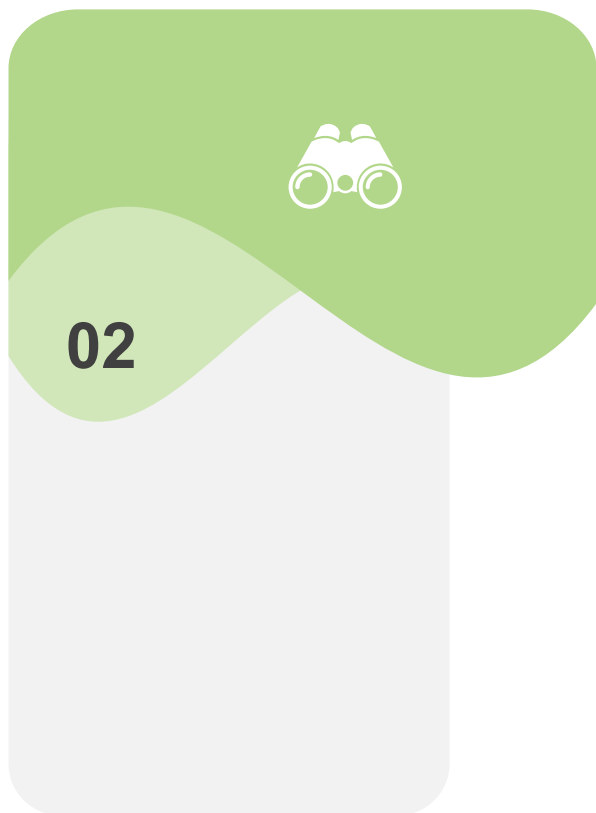
T0+24

Urban : **LOD1** VHR (**1m**).
 Global : Classification in VHR, based on **15 classes** and **change detection**
 Tests sites : 5 cities + 2 regions
 Dev Platform 2sd version

T0+12

Urban : **LOD0** VHR (**1m**) (building/not building)
 Global : Classification in HR (20m), based on **10 classes**
 Tests sites : 1 city (Toulouse)+ 1 region (french PACA)
 Dev Platform 1st version





AI4GEO DEVELOPMENT PLATFORM

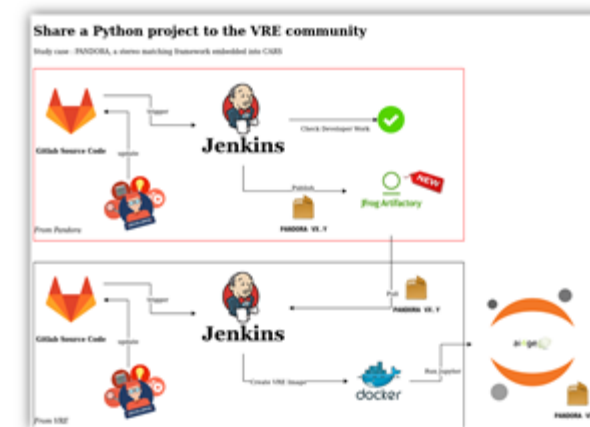
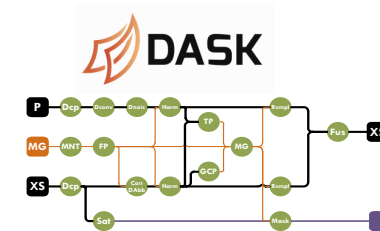
Concepts inherited from previous works (bulk processing)

- Massive image production & scalability
 - Workflow using 2 levels of parallelization (inter node/intra node)
 - Dask used for intra node

- Interoperability and portability (HPC & Cloud)
 - Deployed on CNES HPC + OVH & Orange clouds
 - SafeScale (<https://github.com/CS-SI/SafeScale>) + dockerisation

- Acceleration of development: Agilty + Software factory

- Reference processing framework for new EO missions and platforms

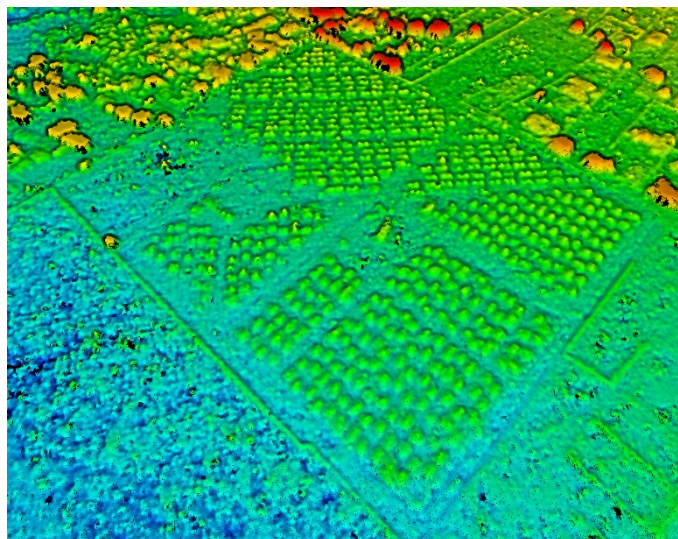


Ex. of massive 3D processing with Satellite Stereo Pipeline on OVH Cloud

- Framework deployed on OVH cluster



- 3D images and DSM from Pleiades tri-stereo (**400km²**)



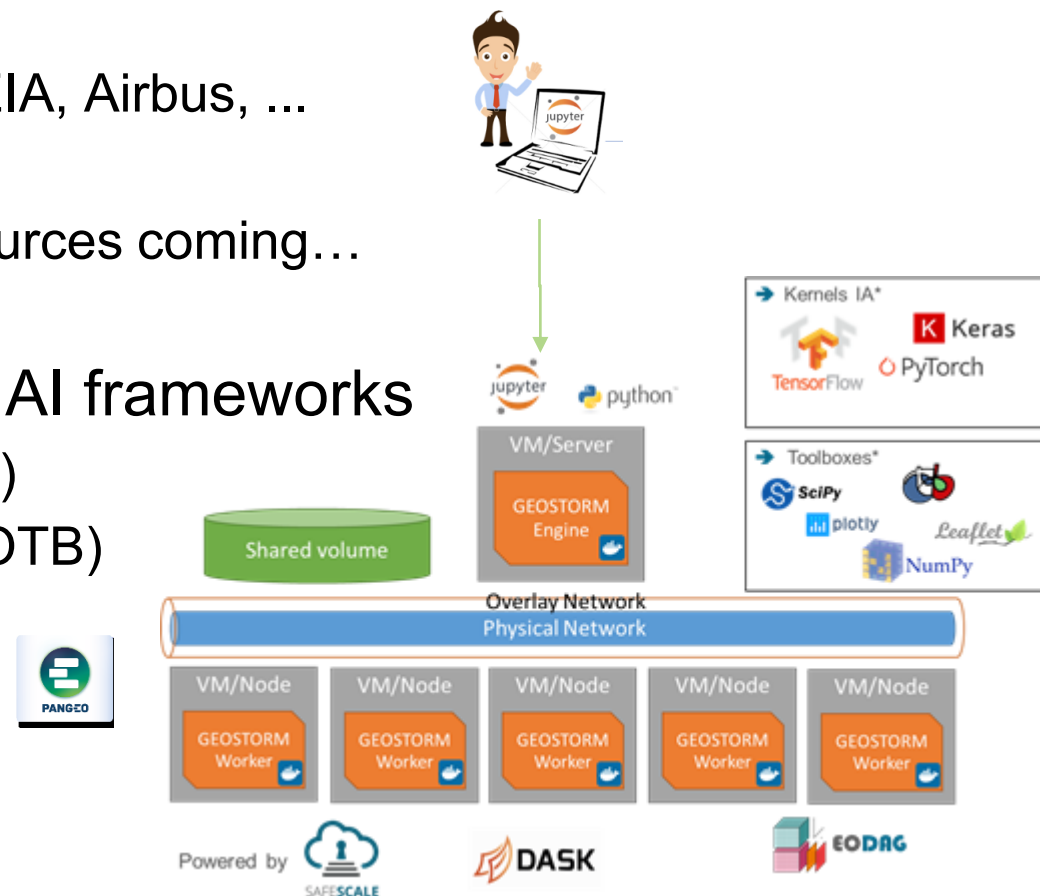
| 16 compute nodes node features | Proc. time |
|-------------------------------------|------------|
| 32 vCPUs, 240 GB RAM, 400 GB SSD | 9 min 10s |

Processing Framework improvement for Analytics

- Multi-sources data access
 - Access to catalogues from Copernicus, THEIA, Airbus, ...
 - EODAG (<https://pypi.org/project/eodag/>)
 - Connectors to OpenStreetMap and other sources coming...

- Foster the use of image processing and AI frameworks
 - Integration with PanGeo (DASK, XARRAY...)
 - Embed IA Kernels & processing toolboxes (OTB)

- From prototype to operational
 - interactive mode : JUPYTER
 - web native and virtual desktop (noVNC), IDE (VS Code)



Processing Framework improvement for Analytics

- Multi-sources data access
 - Access to catalogues files
 - EODAG (<https://pypi.org>)
 - Connectors to OpenStreetMap
- Foster the use of image processing
 - Integration with PanGeo
 - Embed IA Kernels & products
- From prototype to open source
 - interactive mode : JupyterLab
 - web native and virtual desktop

The screenshot displays a JupyterLab environment with a web browser window showing a 'PRODUCTS SEARCH' interface. The interface includes a map of France with a search box, a dropdown menu for 'Product type (*)' set to 'S2_MSI_L1C', 'Start date' (01/05/2019), 'End date' (08/05/2019), and a 'Max cloud cover 50%' slider. A 'SEARCH' button is visible at the bottom of the search panel.

On the right, the JupyterLab code editor shows the following code and output:

```
[1]: from eodag import EODataAccessGateway
dag = EODataAccessGateway()
product_type = 'S2_MSI_L1C'
footprint = {'lonmin': 0.527343, 'latmin': 42.843751, 'lonmax': 3.691406, 'latmax': 44.715513}
cloud_cover = 50
start, end = '2019-05-01', '2019-05-08'
search_results, estimated_total_nbr_of_results = dag.search(
    productType=product_type,
    box=footprint,
    start=start,
    end=end,
    cloudCover=cloud_cover,
)
```

Output of search plugin

```
[2]: from eodag.api.search_result import SearchResult
results_geojson = SearchResult(search_results).as_geojson_object()

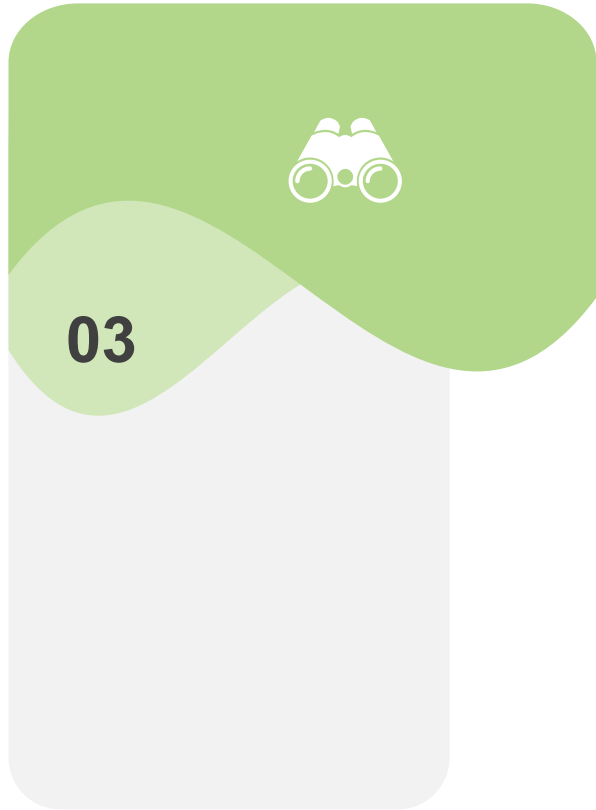
from shapely.geometry import shape, GeometryCollection
features = results_geojson['features']
features = GeometryCollection([shape(feature["geometry"]).buffer(0) for feature in features])

[2]: 
```

Extract products extent

```
[3]: ext = features.bounds
bounds = [[ext[1], ext[0]], [ext[3], ext[2]]]
bounds

[3]: [[41.436336380095, -1.5953369], [45.153837250892, 5.6271142360394]]
```



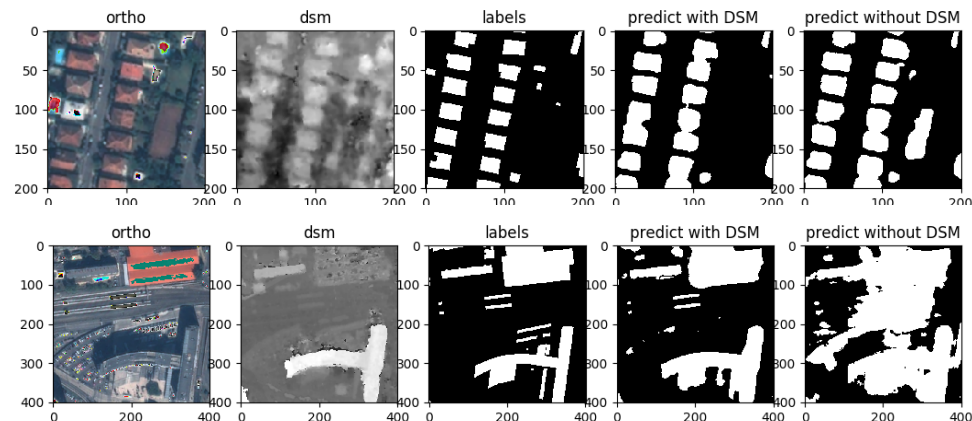
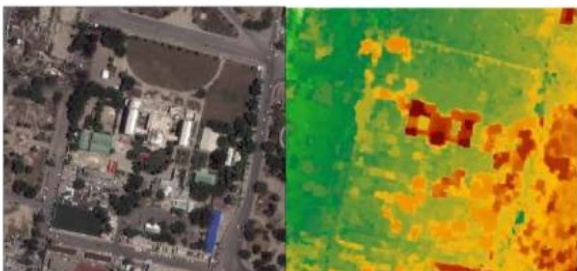
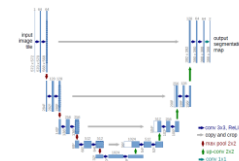
FIRST RESULTS

Example of application : LOD0 in urban areas (1/2)

- Urban mapping, re-constructions progress ...
- Pipeline for building extraction from Pleiades stereo tuples (ortho + DSM)

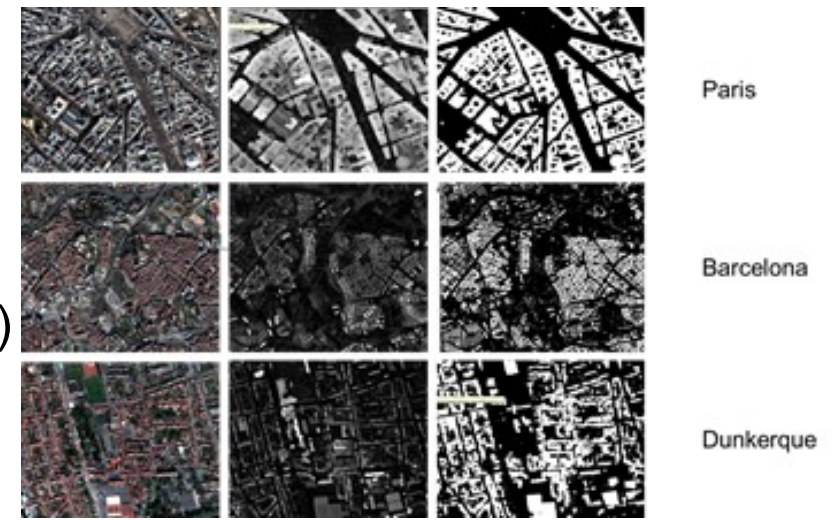


cnes
CENTRE NATIONAL
D'ETUDES SPATIALES
**Stereo
Pipeline**



LOD0 in urban areas (2/2)

- Semantic segmentation
 - Classical U-Net based
 - Learning on Toulouse datasets (OSM + manual labels)
 - Inference on other datasets
 - Parallelisation by tiles 1000x1000



| ID | Accuracy | mIoU | F1Score | Precision | Recall | IoU |
|------|----------|-------|---------|-----------|--------|-------|
| unet | 0,933 | 0,818 | 0,835 | 0,895 | 0,782 | 0,717 |

- Edge regularization
 - Based on GEODAN algorithm
 - Improve to better fit with building constraints (edges oriented, limit nb of right angles, ...)

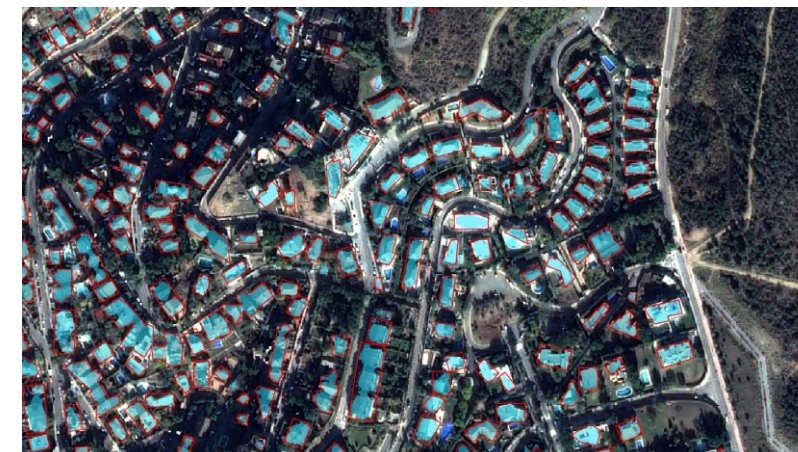


Summary

- A fully integrated environment
 - Facilitate **prototyping AI-based solutions at large scale**,
 - **Data access with connection to existing databases**,
 - Deployable on **HPC or Cloud infrastructures** for massive processing.
- Very promising results on large scale LOD0
 - First release expected June 2020
 - LOD1 preliminary products for Nov 2020.
- Collaborative environment for scientists
 - **Already accessible** on CNES infrastructure,
 - Foster sharing of new methodologies with **labeled datasets**,
 - Promote the usage of **open source toolboxes** such as OTB,
 - Environment for multi-thematics research program = post-docs opportunities open !
- Contact us !



<https://www.ai4geo.eu>



Results LOD0 on Barcelona

ARE YOU AWESOME?

WE'RE HIRING!