Rendering 3D City for Smart City Digital Twin

Lorenzo Adreani[1], Carlo Colombo[2], Marco Fanfani[1,2], Paolo Nesi[1], Gianni Pantaleo[1], Riccardo Pisanu[2]

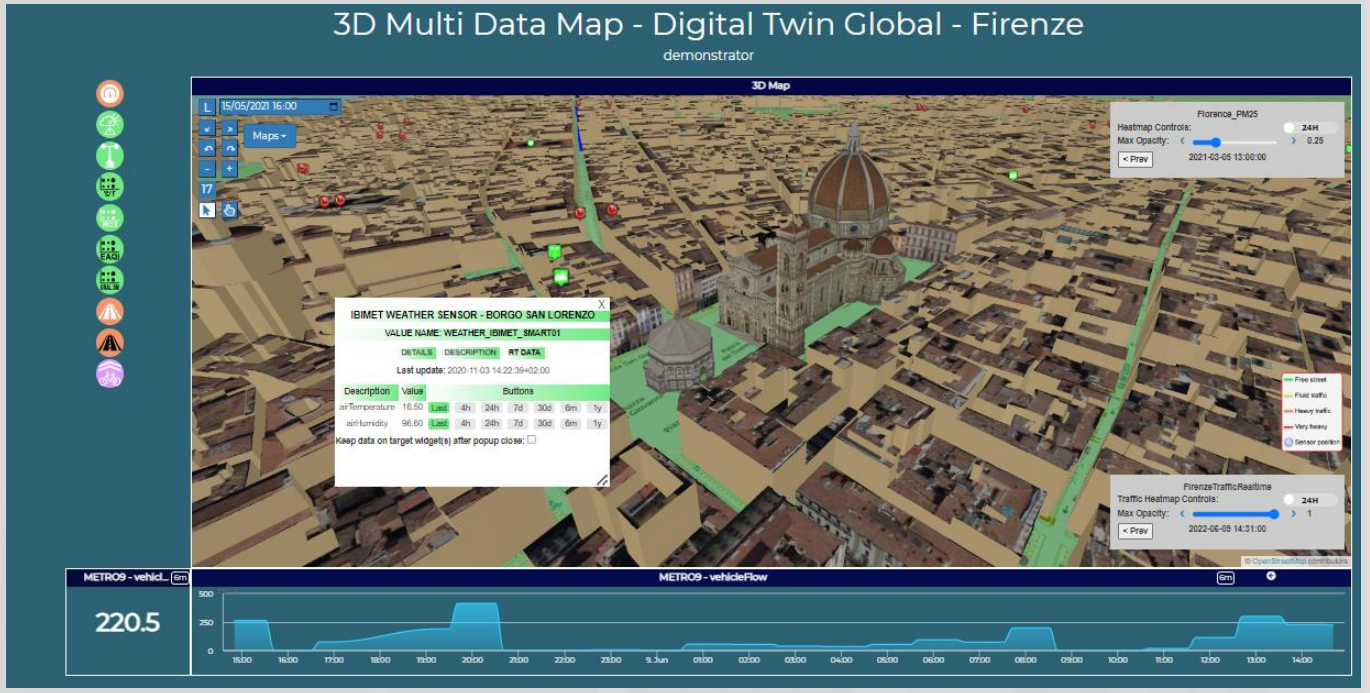
University of Florence, Florence, Italy
email: <name>.<surname>@unifi.it

1) DISIT lab, https://www.disit.org, https://www.snap4city.org
2) Computational Vision Group http://cvg.dsi.unifi.it/cvg/

SMARTCOMP WIP and Demo, June 21 2022

Introduction

- **3D city modelling** is fundamental for representing the city digital twin
- It provides **interactive visualizations** of buildings integrated with IoT data
- In this work a method for producing a 3D city model with **photorealistic rooftop textures,** including sky patterns and terrain elevation**,** is proposed



- Additionally, the 3D city model is integrated into the open-source **Snap4City** framework as a Multi-Data Map with orthomaps, heatmaps, IoT devices, etc. to obtain a **high-fidelity Smart City digital twin**

Requirements

A 3D digital twin must provide:

1. **A 3D representation of buildings** with realistic details in terms of shapes and textures
2. **Ground information,** i.e., shapes and names of roads, squares, and localities
3. **Heatmaps** to represent temperature, traffic flow, pollutant, people flow, etc.
4. **Paths and areas** to describe the perimeters of gardens, cycling paths, border of gov areas, etc.
5. **PINs** indicating the position of services, IoT devices, Point of Interest (POI), Key Performance Indicator (KPI), etc.
6. **Terrain elevation** to realistically represent non-flat areas
7. **Additional 3D entities**, such as trees, benches, fountains, semaphores, digital signages, and any other city furniture

In addition, the solution must provide some interactivity:

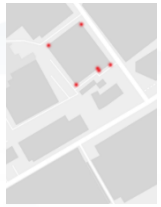
A. Possibility to zoom, rotate, tilt, and pan the scene and change the lighting (day/night)
B. A sky pattern reflecting the actual weather, or weather forecast
C. Providing access to the information associated to PINs
D. Possibility to select a specific building to obtain more detailed information, or move into a BIM view of the building
E. Be able to provoke call backs into business logic tools by selecting specific elements
F. Access to additional 3D underground elements (such as water pipe, subways, etc.) by selecting ground areas

3D City Construction

Multiple images from several points of views

Raw Facades pictures

GIS orthomap



GIS Building plant shapes and positions

Building heights (at the eaves)

Lidar Data (DSM)

Sky Pattern

Terrain (DTM)

Heatmaps, Traffic flow, Pins, IOT, POI, RT data, ...

3D design of High Value Buildings, HVB

Facades patterns extraction

Roof patterns extraction

Create 3D building with flat roofs (by extrusion)

Create 3D buildings with 3D roofs' shapes

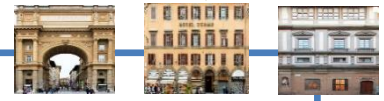
Integrated view of HVB + building with roofs and facades

Generate 3D city representation

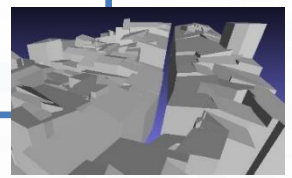
3D City Digital Twin on Dash

HVB model with patterns

Facades' patterns



Roof patterns



Create 3D building with photorealistic texture

3D buildings with roof and facade patterns

Extruded building with picking functionality

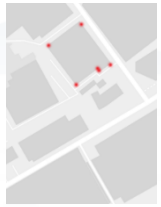
Orientation, Position, Light, zoom, view point, etc.

3D City Construction – in this work

Multiple images from several points of views

Raw Facades pictures

GIS orthomap



GIS Building plant shapes and positions

Building heights (at the eaves)

Lidar Data (DSM)

Sky Pattern

Terrain (DTM)

Heatmaps, Traffic flow, Pins, IOT, POI, RT data, etc.

3D design of High Value Buildings, HVB

HVB model with patterns

Facades patterns extraction

Facades' patterns

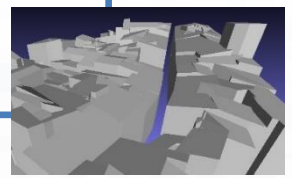
Roof patterns extraction

Roof patterns

Create 3D building with flat roofs (by extrusion)

Create 3D building with photorealistic texture

Create 3D buildings with 3D roofs' shapes



3D buildings with roof and facade **patterns**

Integrated view of HVB + building with roofs and facades

Generate 3D city representation

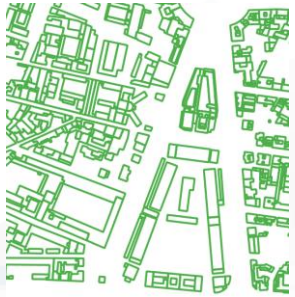
Extruded building with picking functionality

3D City Digital Twin on Dash

Orientation, Position, Light, zoom, view point, etc.

Roof pattern extraction

Orthomaps

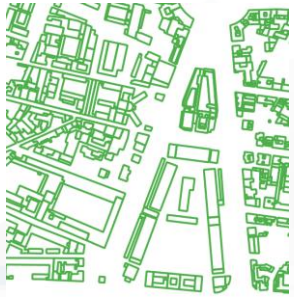


Building shapes

Input

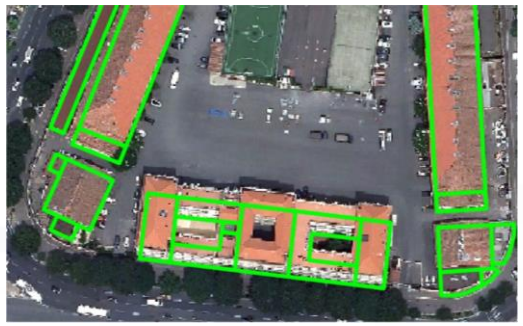
Roof pattern extraction

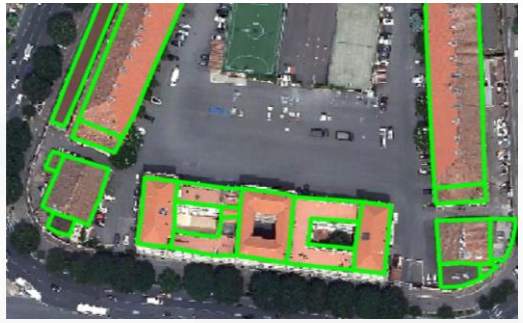
Orthomaps



Building shapes

Input

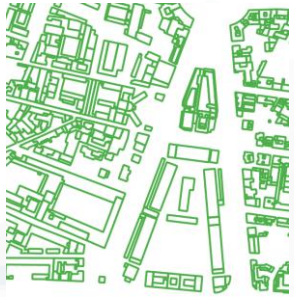




Deep network alignment

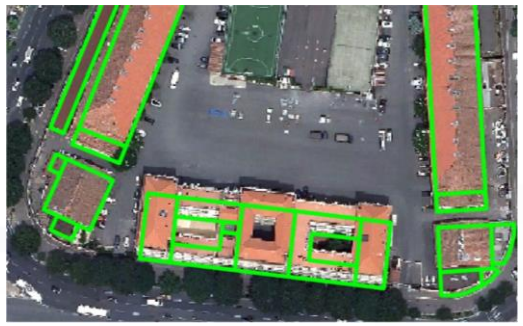
Roof pattern extraction

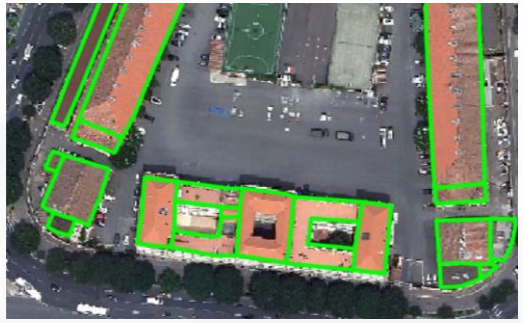
Orthomaps



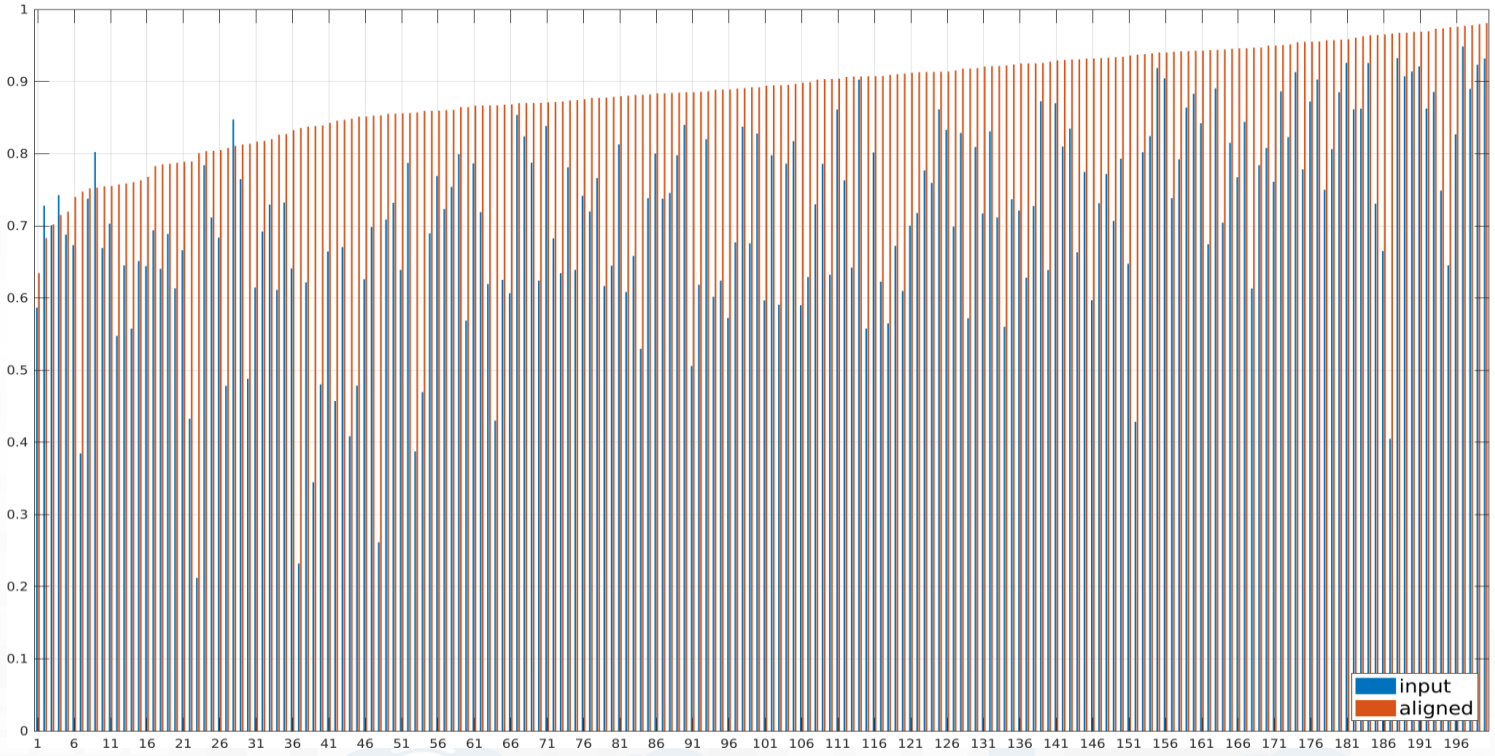
Building shapes

Input





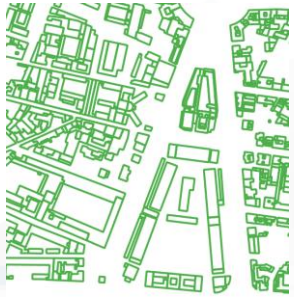
Deep network alignment



After the alignment, **an increases of almost 15%** of the IoU between the building shapes and the orthomap rooftops was measured.

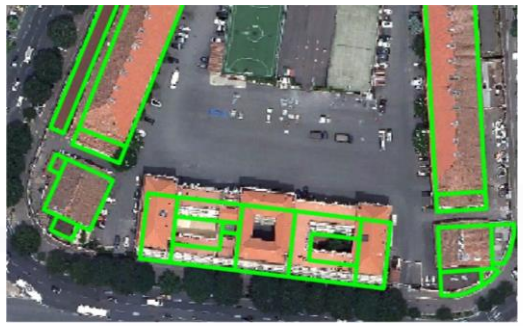
Roof pattern extraction

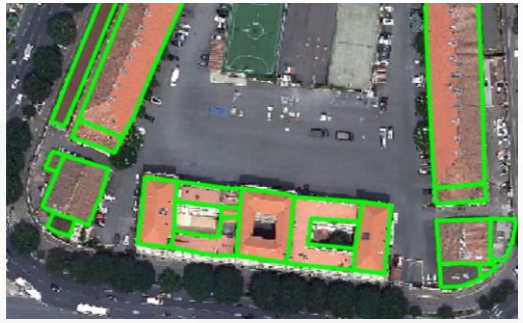
Orthomaps



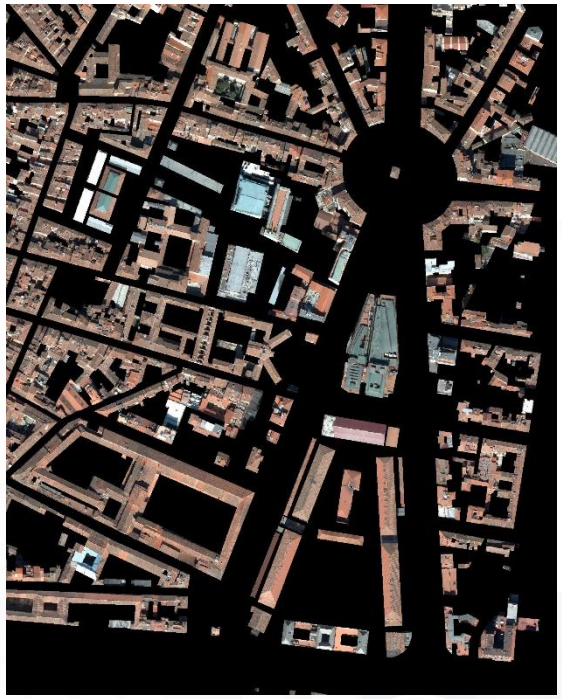
Building shapes

Input





Deep network alignment

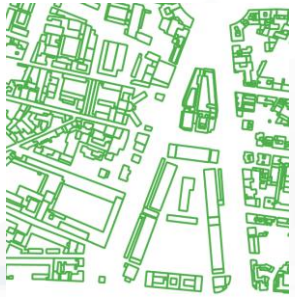


Rooftop texture extraction and warping

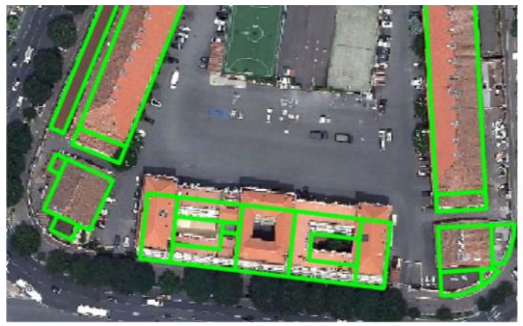
3D Map Texturing

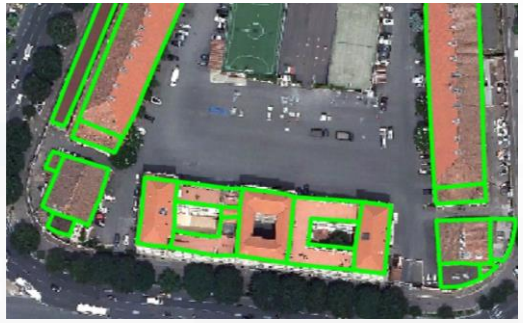
HVB inclusion

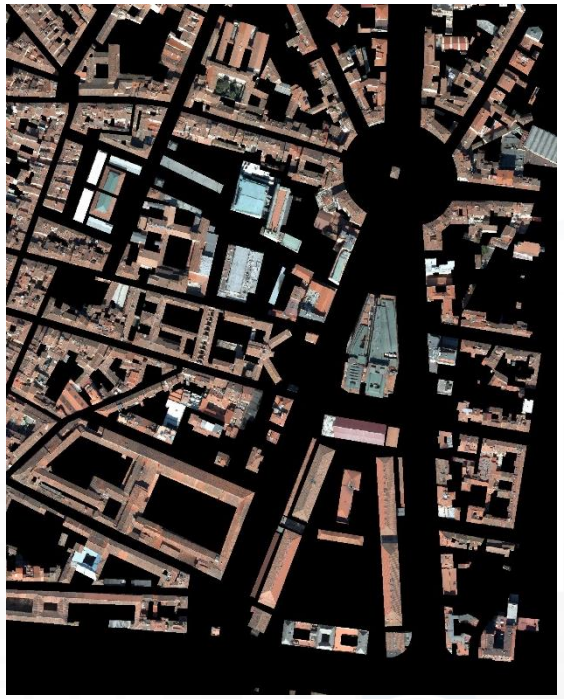
Orthomaps

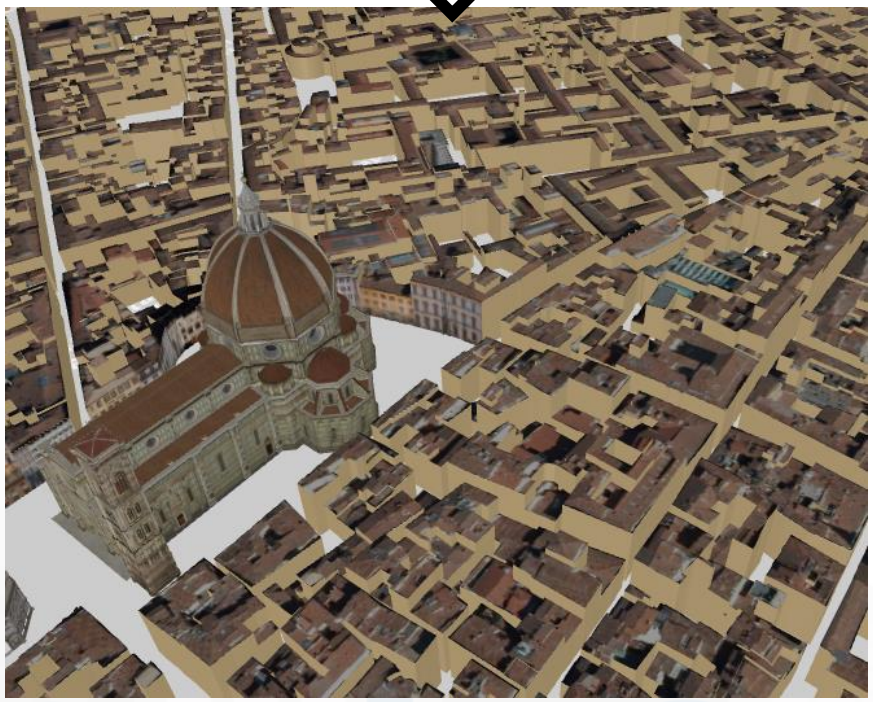


Building shapes





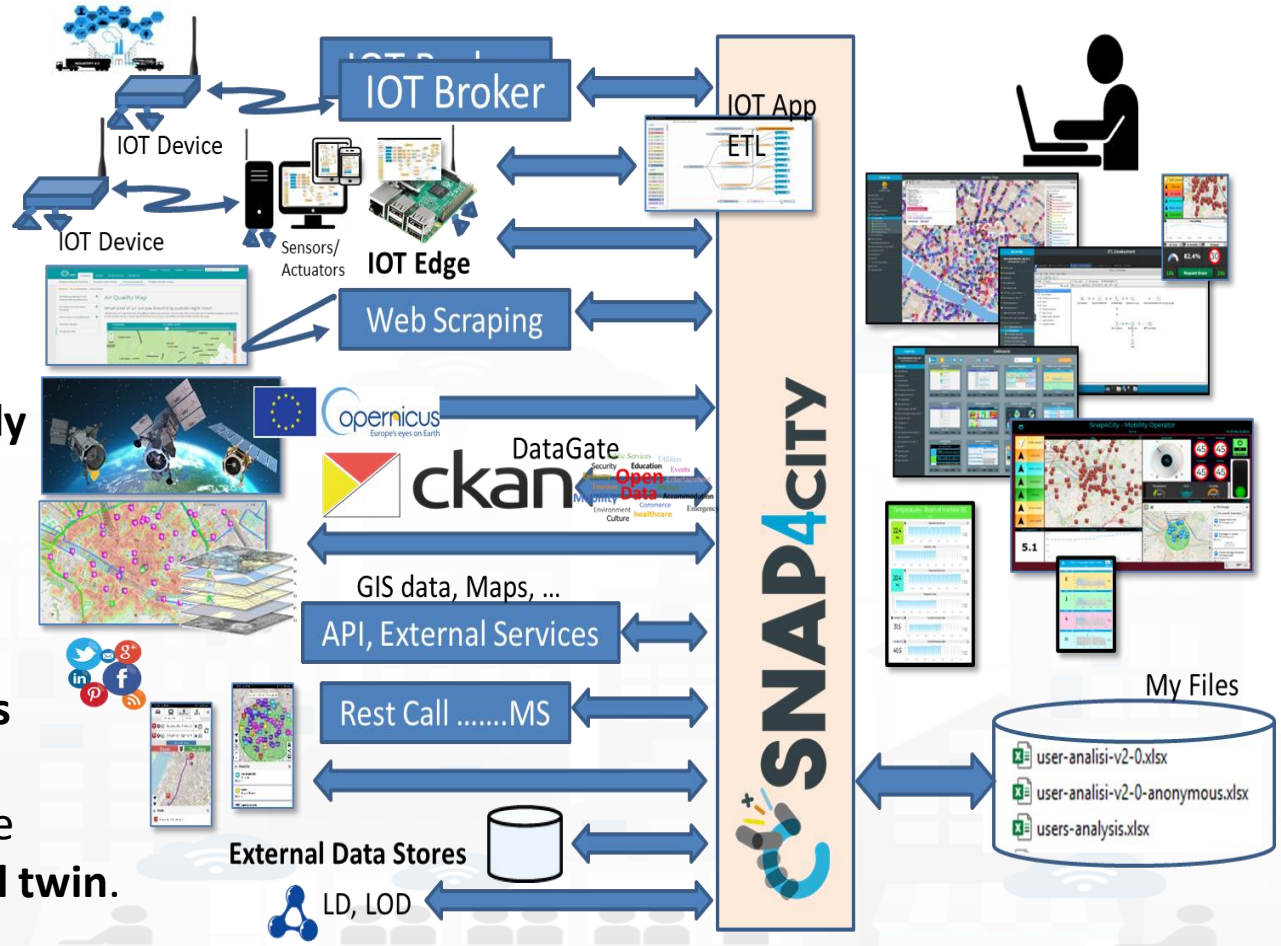




Input → **Deep network alignment** → **Rooftop texture extraction and warping** → **Final textured 3D map**

Snap4City

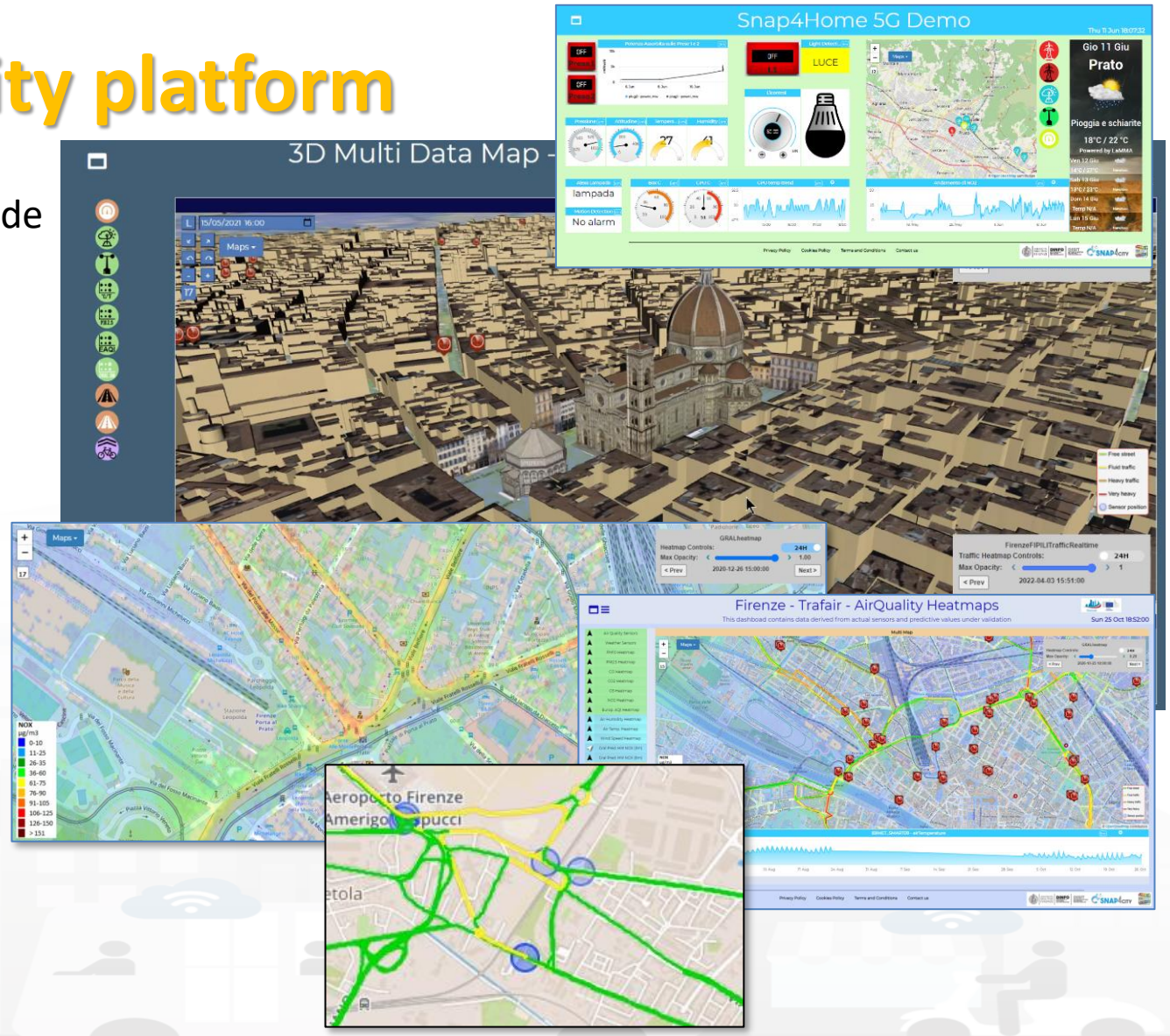
- Snap4City is an **open-source platform** developed at DISIT Lab, University of Florence.
- The platform manages **heterogeneous data sources** (IoT devices, open data, external services, etc.)
- Static and real-time data are **collected and semantically indexed** in an RDF Knowledge Base
- Data retrieved by dedicated APIs and exploited by **Data Analytics processes** and **IoT applications** can be shown to the user through **dashboards** and **widgets**
- Snap4City thus collects, processes, and manages all the data needed to obtain a **high-fidelity Smart City digital twin**.



Integration into the Snap4City platform

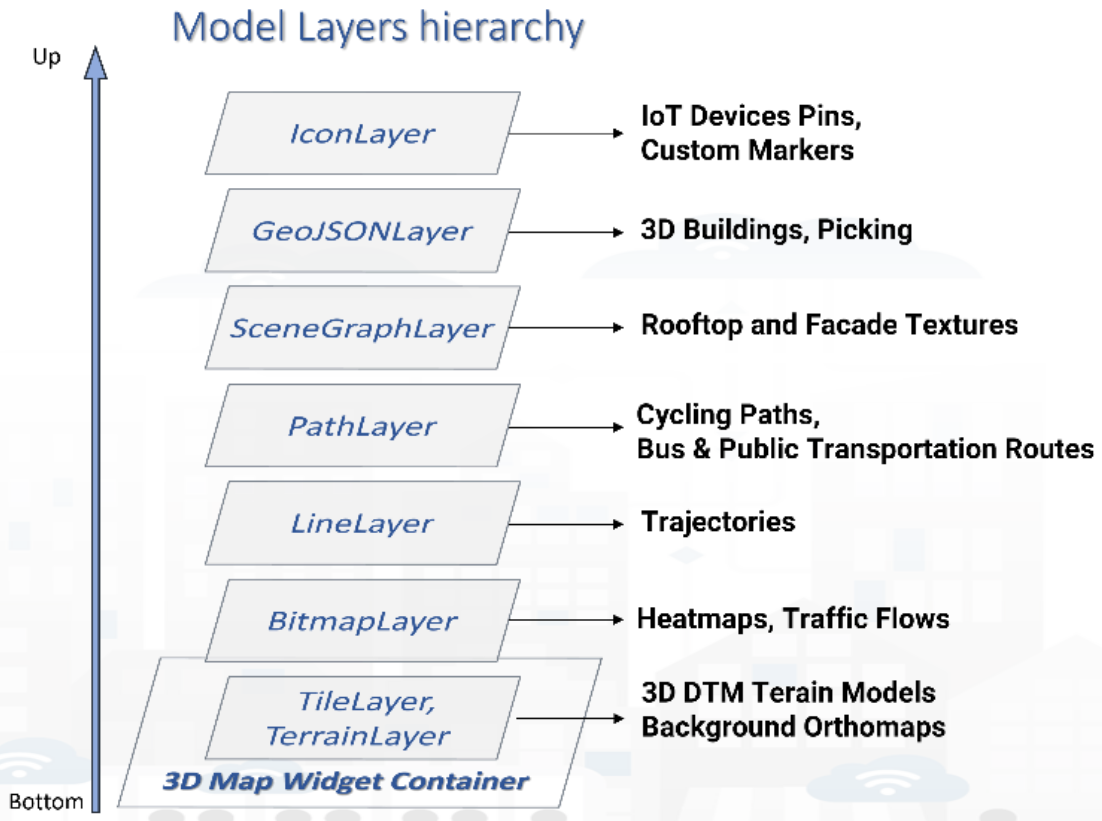
- The 3D city model is integrated into Snap4City to provide a **Multi-Data map**: an interactive 3D environment showing city related data

- The Multi-Data Map incorporates:
  - A DTM based terrain model
  - Generic buildings with roof and façade textures
  - HVB (e.g., Dome, Palazzo Vecchio, etc.)
  - Sky patterns (sunny, cloudy, etc.)
  - Orthomaps and heatmaps (temperature, traffic, pollutant, etc.)
  - Cycling paths, bus routes and traffic flows
  - POI, and IoT sensors with pop-ups showing real time data
  - 3D Pillars reporting values of specific sensors (temperature, traffic flow, people counting, pollutant, etc.)

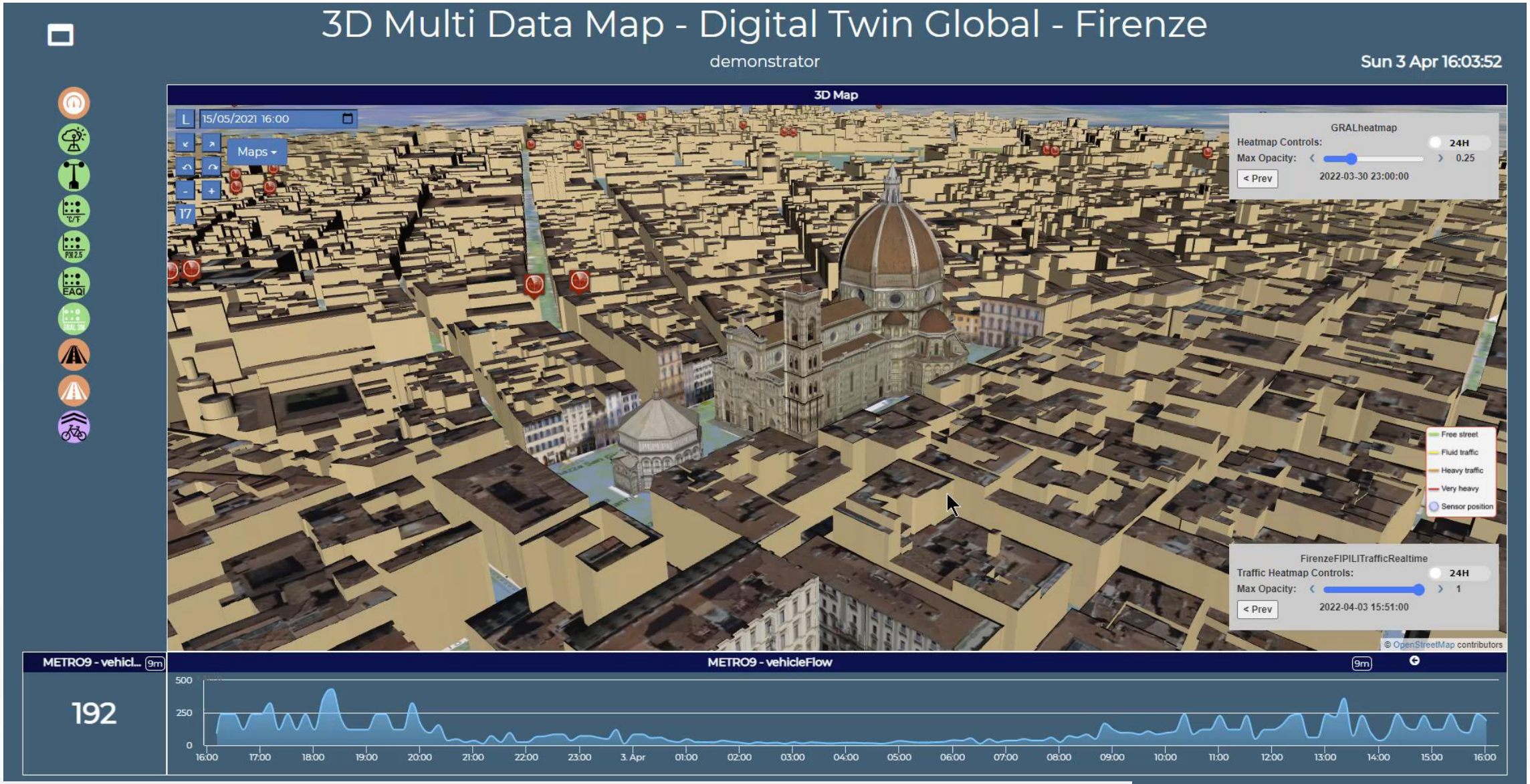


Integration into the Snap4City platform

- The Multi-Data Map 3D city digital twin is created using the **multi-layer structure** of the open-source library deck.gl
  - Tiled **TerrainLayers** were used to create a 3D terrain model textured with orthomaps
  - A **BitmapLayer** was used to display heatmaps and traffic flows retrieved from a geo-server
  - Paths and geometries were displayed using **LineLayer** or **PathLayer**
  - LoD1 3D buildings (extruded from GoeJSON files) were included though a **GeoJSONLayer**
  - Textured 3D buildings and HVB (saved as glTF or GLB files) were loaded using a **SceneGraphLayer**.
  - An **IconLayer** is implemented to represent the IoT devices with specialized icons for each device category



- All the layers can be **loaded at runtime** on user demand.



https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MjUxMA==

Conclusions and future work

- An approach for 3D city modeling was proposed in order to obtain a high-fidelity Smart City digital twin
- In particular, we presented a method for rooftop pattern extraction to be used to enrich 3D building models with photorealistic textures
- Additionally, the obtained 3D city model was integrated into the open-source Snap4City platform, including also terrain elevation, sky pattern, and static and dynamic information coming from IoT devices, open data, and external resources
- Future work will encompass the development of a method for automatic façade texturing, as well as the construction of building models with detailed 3D roof structures exploiting Lidar based DSM data
- The code of the open-source Snap4City Dashboard Builder is available at the following GitHub repository: https://github.com/disit/dashboard-builder