Expanding Digital Design Workflows with Geospatial Analytics: Linking Grasshopper3D with Google Earth Engine

Iacopo Neri[1,2]

[1]Digital Visual Studies, University of Zurich, Zurich/Switzerland · iacopo.neri@uzh.ch
[2]Advanced Architecture Group, Institute for Advanced Architecture of Catalonia, Barcelona/Spain

Abstract: The following body of work introduces a plugin that links the visual scripting language Grasshopper3D (GH) to the Google Earth Engine (GEE) in order to easily fetch geospatial information relative to various societal issues and for any geographical area under study, inside the Rhino modelling software.
Aiming at expanding the field of Digital Landscape Architecture with novel content to analyse and design, it provides designers with more than thirty years of historical imagery and scientific datasets, collected in GEE on a daily basis by several institutions around the world. Leveraging the intuitiveness of the visual scripting language, it computationally empowers designers with geospatial insights without the need for any GIS skills and has been proved a successful platform for teaching purposes. Encouraging learning through application, the paper discusses three teaching experiences, which adopted the proposed tool to visualise river dynamics in time, resource-specific maps of land consumption for cities and street-sensitive accessibility maps through the additional integration of OpenStreetMap data.

Keywords: Algorithmic design and analysis Landscape, teaching digital landscape architecture, UAV imagery and Remote sensing in landscape architecture

1 Introduction

During the last three decades, a constellation of computational applications has emerged that empowers architects and designers to respond to the challenges of the AEC and planning sectors through highly technological and innovative means. Active since late 2007, the Rhinoceros's visual programming language: Grasshopper3D (GH), has been proved to be among the most successful examples of this kind. It has offered a visually-intuitive medium to teach and compute advanced computational pipeline without writing one line of code, and has become an asset for both the academic and the industrial realms (CASTELO-BRANCO & LEITÃO 202). Additionally, whether to simulate microclimatic conditions (MACKEY et al.. 2017), calculate structural performances (PREISINGER & MORITZ 2014), or work with georeferenced data (DOGAN et al. 2018) to name a few, over the years GH has collected an extensive amount of plugins, developed by an active community of computational designers, to expand its influence in many aspects of the design process and in relationship to the many actors involved. Finally, by reshaping the traditional drawing tools through mathematics and functions, it has provided designers with novel ideas to design while deeply influencing all scales of the project, from Digital Fabrication to Digital Landscape Architecture. Placed within this line of investigation, the following body of work introduces a plugin that links Grasshopper to Google Earth Engine to instantly fetch selected geospatial layers for any geographical area of interest. In this sense, it enables designers to access spatial insights related to more than thirty years of remote sensing data, collected by a plethora of satellites and processed by research institutions from all over the world. Answering to the call for data democratization

Journal of Digital Landscape Architecture, 8-2023, pp. 443-450. © Wichmann Verlag, VDE VERLAG GMBH · Berlin · Offenbach. ISBN 978-3-87907-740-3, ISSN 2367-4253, e-ISSN 2511-624X, doi:10.14627/537740047.

while rendering large-scale computing accessible to non-experts, the Google Earth Engine (GEE) is a "cloud-based platform for planetary-scale geospatial analysis [that seamlessly gives to] not only traditional remote sensing scientists, but also a much wider audience, [access to many] societal issues including deforestation, drought, disaster, disease, food security, water management, climate monitoring and environmental protection" (GORELIK et al. 2017). For this reason, GEE is built around a petabytes-large catalogue of georeferenced data and an Application Programming Interface (API) to access and process server-side the same layers; achieving in this manner high speed performances and becoming suitable not only for global-scale calculation processes, but also for more explorative and experimental approaches, common in design processes.

2 The Toolkit

As an entry point for designers to explore geospatial analytics through Google Earth Engine, the toolkit proposed consists mainly of five GH components to import, spatialize, process and calculate geospatial raster layers through the Earth Engine API Python library[1] and via Hops. Being a recent development in the Grasshopper3D suite, Hops is the first package to efficiently link the visual scripting tool to the real potentialities of the Python programming language. By externally running CPython code via a Flask application, it allows Python scripts to be implemented in GH unconstrained by the limitations of predefined libraries and open to the vastness of community-driven packages available online.

Being one of these contributions, the Earth Engine API is the official Python client library to dynamically access GEE. Used within the tailored Flask application, it runs requests from the inputs in GH to the online GEE server and consequently fetches the required information. More precisely, the discussed custom components to connect GH to GEE are:

1) *ee_image*: it permits the download of images from GEE for any geographical area of interest and functions as the primary component to explore GEE.
2) *ee_imageColl*: it enables to work with the more advanced imageCollection typology and, compared to the ee_image, requires extra information in respect to the date, or permitted cloud coverage to consequently extract images.
3) *ee_ND*: it engages with GEE to create normalised difference indicators on the server side by providing multiple bands to work with, and functions as an entry point to the world of remote sensing indicators
4) *ee_cumCost*: it calculates cumulative cost analysis, which are commonly used to spatialize accessibility, provided a cost to travel over a territory and an initial set of origins
5) *reproject_UTM*: an utility component to manage coordinate reference systems and align data fetched from GEE with the outputs of other plugins for GIS operations

The requirements to use the aforementioned components are kept very concise and focus on providing the maximum flexibility with the minimum amount of inputs, and always comprises: an area of interest to download the image from, a resolution -in metres- to balance the amount of information to be downloaded, and the layer, with respective bands, that we are interested in accessing. Additionally, more inputs can be requested to calibrate the functions

[1] Earthengine-Api: Earth Engine Python API. Accessed January 2023. http://code.google.com/p/earthengine-api/.

of the specific components, like in the case of the *ee_ND* that requires more than one band to reciprocally subtract, or the *ee_cumCost* which requires locations of origin for the accessibility analysis to be calculated from. This being said, the toolkit automatizes a series of spatial operations common in Geographical Information Systems (GIS) to deal with raster layers, like is the case of resampling operations to obtain custom resolutions, or mathematical operations to calculate remote sensing indicators. It is important to notice that it does so in the background, opening up possibilities for the users to interact with the tool only through specifically opinionated inputs in order to facilitate its generic usability and versatility. In this sense, the tool aims at providing a simplified pipeline for computational designers to investigate the immensity of the Google Earth Engine database for design purposes through a scarce set of inputs, allowing them to obtain material for further analysis and manipulations with conventional processes – through standard components in Grasshopper3D – in a fast, and interactive fashion and without the need to be GIS experts.

Finally, the open source nature of the tool – a Python Flask application – permits an in-depth customization of each component – if equipped with enough knowledge of the Python programming language – and it has been proved to be a fruitful case to learn and apply computational logics in a pedagogical sense. It balances levels of complexity when approaching GIS processes through visual programming while maintaining the possibility to read and study the back-end codes when necessary.

3 Learning Through Application

Through one year of teaching experience, the proposed plugin has been tested on several occasions and has been proved to be versatile enough for different case studies, enhancing in a broad sense the toolset that designers possess when approaching a territorial project – for visualisation or analytical purposes – and not focusing on highly specific outputs. In the following section, the paper discusses three such occasions where the methodology has been shared with students to analyse river patterns in time, resource-specific maps of land consumption for cities, and street-sensitive accessibility maps through the integration of OpenStreetMap data relative to high resolution street networks.

More precisely, the case studies hereby collected are the results of the Geomining lecture for the Master in Landscape Urbanism at the Architectural Association School of Architecture in London, the Earthy Indexes workshop at the CAADRIA conference 2022/23, and the one-week Urban Analytics workshop for the IAAC Global Summer School 2022.

3.1 Analysing River Patterns in Time

The analysis of river patterns in time is an important tool for understanding the dynamics of river systems and the impacts of natural and human-induced changes on these systems. Used to inform a wide range of decisions related to the management and protection of river systems and the resources they provide, such as flood risk assessment, water resource management, environmental impact assessment, and land use planning, it is a fundamental asset for Landscape Architecture, which usually requires tedious data research and modelling.

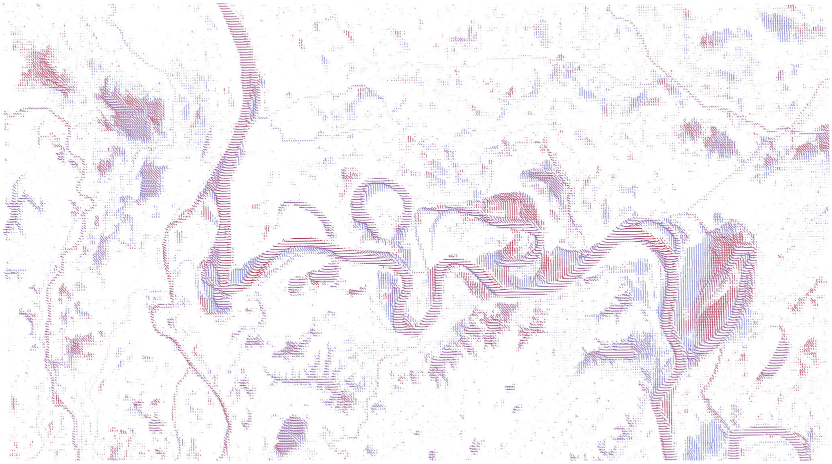

Fig. 1: Visualisation for the Yangtze river as class material for the Geomining lecture led by Iacopo Neri at the Architectural Association School of Architecture, 2021/22. It shows presence, recurrence and seasonality of hydrologic patterns, respectively by means of length, colour and angle of each single line.

Unequipped with a specific plugin, these studies are commonly carried out in GH through ad-hoc scripting via iterative logics, such as for river meandering and oxbow lake simulation[2] or water runoff studies which can be used as support material. Despite being excellent case studies to teach iterative logics and loops (i. e. using the Anemone plugin[3]), they often fail to reach a high level of specificity and end up in the realm of design exercises compared to territorial studies: fruitful to inspire design processes but insufficient for serious analytical purposes.

On the other hand, there are several methodologies that can be used to map river dynamics in time using GIS, including time-enabled data, dynamic modelling and finally time series analysis. Despite being able to provide highly specific and precise assessment of river dynamics, studies via time-enabled data or computational models are generally expensive or demanding to implement due to the requirements of on-site equipment or highly trained professionals to set up and run hydrologic models. On the contrary, remote sensing has been widely used in the analysis of river patterns in time over the past few decades as it allows for the collection of large amounts of data over a broad area in a relatively short period of time; permitting a wide range of spatial and temporal scales to be included in a interoperable medium for the experts of the field to disseminate the results of their research.

2 For an example of river meandering and oxbow lake simulation using iterative logics, see "*Flowing Towards*" presented by the author and Erzë Dinarama during the Assume there is a Landscape exhibition (Lambro, Italy). Accessed January 2023. https://assumetheresalandscape.com/E2107.

3 Anemone plugin by Mateusz Zwierzycki. Accessed January 2023. https://www.food4rhino.com/en/app/anemone.

In this sense, the JRC Global Surface Water mapping layer[4] offers an unprecedented synthetic image of more than 4 million scans from Landsat 5, 7, and 8 to describe in high-resolution the long-term changes happening in river systems from early 1984 until the end of 2021. Hosted in GEE as a multi-band 30m resolution image, it can be easily queried via the proposed *ee_image* component to visualise for any area of interest the patterns of *extension*, *seasonality* and *recurrence* of water to name a few. These layers precisely have been class material during the Geomining lecture at the Architectural Association School of Architecture where participants drew a synthetic line-map of temporal river dynamics (Figure 1) almost-instantly and without geographical restraints. Taking advantage of the extensive representational possibilities of GH and adopting colour, angle and length as parameters, the map reported not only where it is possible to find water resources, but also their permanent loss and yearly frequencies, thus providing a wider understanding of the ephemerality of water compared to a standard layer by layer visualisation. Additionally, and only thanks to the implemented pipeline, no particular download was required to compute the analysis. Avoiding to redundantly download entire databases by running area-specific queries in GEE is far more than secondary as it prevents common issues concerning not only memory availability but also computational power and computing times on the designer's machine.

3.2 Maps of City Consumption

The majority of people in the world live in cities, which currently only take up 3% of the Earth's surface but have transformed 70% of the planet through human activities (CIESIN 2016). In this sense, cities around the world are interconnected and constantly exchanging resources, but the traditional link between places of consumption and places of extraction that was once vital for a city's prosperity has been disrupted. As geographical proximity became less important for urban success, the environmental impact of this shift was overlooked, contributing to the unsustainable nature of modern society, particularly due to the physical separation of consumption and resource extraction.

Aiming to shorten the awareness gap that current planetary urbanisation has produced, the Earthy Indexes workshop held by the author and Erzë Dinarama at the CAADRIA conference 2022/23 presented a computational methodology – strongly supported by the discussed pipeline – to engage with the concept of ecological footprint at the city scale. More specifically, it challenged participants to crossread resource-specific demands of agricultural, pasture or forest land with context-specific land availability and land accessibility; finally envisioning up to which extension a city would consume if operating only by proximity logics (Figure 2).

In line with another study on *Spatialized Metropolitan Ecological Footprints* (Neri 2021), the analysis computes pro-city land consumption values to fulfil the annual demand of a selected resource for its entire population and consequently queries exact amounts of land, filtered and ranked by its infrastructural accessibility. It exploits mainly two data layers: the GlobCover 2009[5] for a 300 m resolution global land cover map, and the Oxford's Global Friction Surface 2019[6] layer to feed a territorial road-sensitive cumulative cost analysis via

[4] Accessed in Google Earth Engine as "JRC/GSW1_4/GlobalSurfaceWater", January 2023.
[5] Accessed in Google Earth Engine as "ESA/GLOBCOVER_L4_200901_200912_V2_3", January 2023.
[6] Accessed in Google Earth Engine as "Oxford/MAP/friction_surface_2019", January 2023.

the *ee_cumCost* component. More specifically, the latter offers a map where every pixel is given a speed to travel based on the local road infrastructure at approximately 900 m scale and based on a combination of national and global (OSM) data.

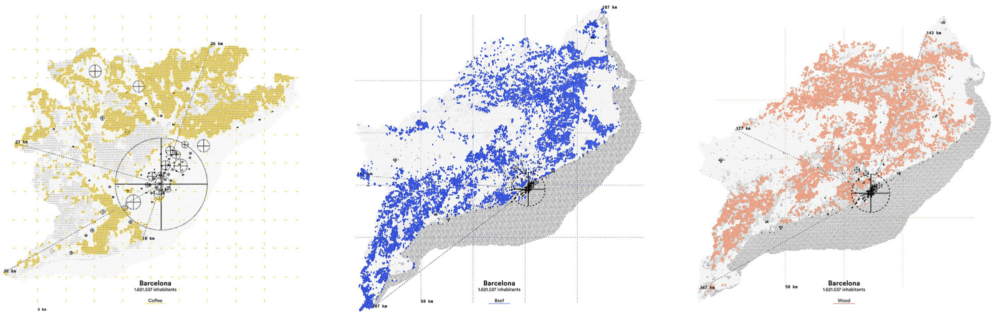

Fig. 2: Ecological footprint studies for coffee, beef and wood, for the city of Barcelona as part of the Earthy indexes workshop led by Erzë Dinarama and Iacopo Neri at the CAADRIA 2022 – Post Carbon conference

Bridging statistical data (e. g., *demography* and *land consumption values*) with geographical data (e. g., *land use* and *accessibility maps*), this approach offers a fruitful pedagogical platform to engage with territorial indexes, while discussing the role of critical cartography in support of sustainability-related studies.

3.3 Urban Accessibility Maps

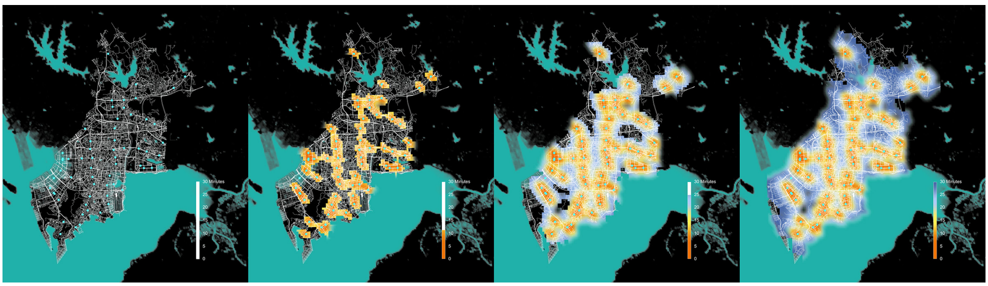

Fig. 3: Street-sensitive isochrone study through Google Earth Engine and Open Street Map for the city of Shenzhen as part of the IAAC Global Summer School's Urban Analytics workshop led by Iacopo Neri and Eugenio Bettucchi

Finally, the proposed pipeline has been adopted to study micro-scale mobility patterns. Reflecting on the aforementioned Oxford Global Friction Surface layer, the *ee_cumCost* component offers the possibility to alternatively use an ad-hoc friction layer to run cumulative cost analysis, therefore, exploiting GEE only for computing purposes and not for data collection. Again, OSM data provides a valuable medium to fulfil this goal, and can be easily accessed in GH via many workflows (i. e. Urbano[7], Gismo[8]) and geographically aligned with the proposed pipeline via the utility *reproject_UTM* component. Technically, the cumulative cost component welcomes any sort of curve-based geometry to paint an image on the GEE server with custom values and for any provided resolution, permitting the modelling of district-scale isochrone studies unlimited by the coarser standard friction layers of GEE (Figure 3).

This was the subject of the one-week Urban Analytics remote workshop for the IAAC 2022 summer school led by the author together with Eugenio Bettucchim, where the international audience of participants mapped for their home-towns a series of accessibility maps to various amenities and public services, collectively discussing by comparison the manifold forms of the x-minutes city.

Other scholars have been using OSM data to create intuitive pipelines for accessibility studies via network graph (Geoff 2020). Despite being widely used in mobility studies as an excellent medium to represent complex relationships between the different elements of a street network (i. e. intersections, roads, and traffic flow), graph modelling requires extensive cleaning operations in its set up phase, which – for a crowd-sourced and in-development database such as OpenStreetMap – may disincentive non-expert users in comfortably interact with the algorithm. Trading specificity over usability, the proposed pipeline suggests a raster based approach to model street-sensitive accessibility maps, solving the incongruencies within the OSM network with a choice of pixel-resolution.

4 Conclusion and Outlook

In conclusion, the Grasshopper3D addon discussed in this text allows designers to access and utilise geospatial data from the Google Earth Engine platform in their design process. The plugin consists of five components that import and process geospatial data through the Earth Engine API Python library and Hops. The Earth Engine API is the official Python client library for accessing GEE and is used within a tailored Flask application to fetch the required information in response to inputs from the GH components. These components allow designers to explore and use GEE data, specifically imagerial data, for various purposes and scales: from digital fabrication to digital Landscape Architecture, and integrating it with more traditional computational pipelines.

As proved through one year of teaching experience, this plugin represents a valuable tool for designers seeking to use geospatial data in their work and expands the capabilities of GH by linking it to GEE's extensive data catalogue and powerful processing capabilities.

Further steps can be taken to extend the plugin with GEE’s Machine Learning algorithms like the ones used for classification, clustering, regression or feature extraction, to name a few. Related to a higher level of expertise, these algorithms allow to reproduce at will many

7 Urbano plugin by Timur. Accessed January 2023. https://www.food4rhino.com/en/app/urbano.

8 Gismo plugin by A. Di Nunzio, D. Spasic, G. Meunier, M. Venot. Accessed January 2023. https://github.com/stgeorges/gismo.

of the pre-processed layers of GEE, which might be used to accomplish ad-hoc or higher-resolution maps, similarly to the example of the district-scale isochrone studies via externally fetched OSM data, as well as to include design inputs in the forecast of their impacts.

Acknowledgements

The author would like to express his warmest gratitude to Erzë Dinarama and Eugenio Bettucchi for their significant contributions in the form of comments and suggestions. The shared teaching experiences provided an invaluable opportunity to evaluate the adaptability of the approach, reinforce key parts of the methodology, and venture into novel research directions.

References

BOEING, G. (2020), A Multi-Scale Analysis of 27,000 Urban Street Networks: Every US City, Town, Urbanized Area, and Zillow Neighborhood. Environment and Planning B: Urban Analytics and City Science, 47 (4), 590-608. doi:10.1177/2399808318784595.

CASTELO-BRANCO, R. & LEITÃO, A. (2022), Comprehending Algorithmic Design. In: Computer-Aided Architectural Design. Design Imperatives: The Future Is Now, 1465, 15-35. doi:10.1007/978-981-19-1280-1_2.

CENTRE FOR INTERNATIONAL EARTH SCIENCE INFORMATION NETWORK (2016), Land and Water Area, v4: Gridded Population of the World (GPW), v4 | SEDAC. https://beta.sedac.ciesin.columbia.edu/data/set/gpw-v4-land-water-area (Jan. 2023).

DOGAN, T., SAMARANAYAKE, S. & SARAF, N. (2018), Urbano: A New Tool to Promote Mobility-Aware Urban Design, Active Transportation Modeling and Access Analysis for Amenities and Public Transport. In: Proceedings of the Symposium on Simulation for Architecture and Urban Design, 1-8. SIMAUD '18. Society for Computer Simulation International, San Diego, CA, USA.

GORELIK, 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, Big Remotely Sensed Data: tools, applications and experiences, 202 (December), 18-27. doi:10.1016/j.rse.2017.06.031.

MACKEY, C., GALANOS, T., NORFORD L. & ROUDSARI, M. S. (2017), Wind, Sun, Surface Temperature, and Heat Island: Critical Variables for High-Resolution Outdoor Thermal Comfort. doi:10.26868/25222708.2017.260.

NERI, I. (2021), MIDA Metropolitan Impacts and Drivers Assessment. In: Metropolitan Landscapes: Towards a Shared Construction of the Resilient City of the Future, 173-182. doi:10.1007/978-3-030-74424-3_13.

PREISINGER, C. & HEIMRATH, M. (2014), Karamba – A Toolkit for Parametric Structural Design. Structural Engineering International, 24 (2), 217-221. doi:10.2749/101686614X13830790993483.