Comparative Analysis of Geospatial Tools for Solar Simulation

León-Sánchez, Camilo; Giannelli, Denis; Agugiaro, Giorgio; Stoter, Jantien

DOI
10.1111/tgis.13296

Publication date
2025

Document Version
Final published version

Published in
Transactions in GIS

Citation (APA)
León-Sánchez, C., Giannelli, D., Agugiaro, G., & Stoter, J. (2025). Comparative Analysis of Geospatial Tools for Solar Simulation. *Transactions in GIS*, *29*(1), Article e13296. https://doi.org/10.1111/tgis.13296

Transactions in GIS

WILEY

Transactions in GIS

RESEARCH ARTICLE OPEN ACCESS

Comparative Analysis of Geospatial Tools for Solar Simulation

Camilo León-Sánchez[1] | Denis Giannelli[2] | Giorgio Agugiaro[1] | Jantien Stoter[1]

[1]Department of Urbanism, Delft University of Technology, Delft, The Netherlands | [2]Digital Solutions Team – Energy and Environment Division, Sweco Belgium NV, Antwerp, Belgium

Correspondence: Camilo León-Sánchez (c.a.leonsanchez@tudelft.nl)

Received: 14 August 2024 | **Revised:** 17 October 2024 | **Accepted:** 17 December 2024

Funding: The authors received no specific funding for this work.

Keywords: ArcGIS pro | CitySim | GRASS GIS | Ladybug | SAGA GIS | SimStadt | solar irradiation analysis | UMEP

ABSTRACT

This paper performs, describes, and evaluates a comparison of seven software tools (ArcGIS Pro, GRASS GIS, SAGA GIS, CitySim, Ladybug, SimStadt, and UMEP) to calculate solar irradiation. The analysis focuses on data requirements, software usability, and accuracy simulation output. The use case for the comparison is solar irradiation on building surfaces, in particular on roofs. The research involves collecting and preparing spatial and weather data. Two test areas—the Santana district in São Paulo, Brazil, and the Henio rural area in Raalte, the Netherlands—were selected. In both cases, the study area encompasses the vicinity of a weather station. Therefore, the meteorological data from these stations serve as ground truth for the validation of the simulation results. We create several models (raster and vector) to meet the diverse input requirements. We present our findings and discuss the output from the software tools from both quantitative and qualitative points of view. Vector-based simulation models offer better results than raster-based ones. However, they have more complex data requirements. Future research will focus on evaluating the quality of the simulation results on vertical and tilted surfaces as well as the calculation of direct and diffuse solar irradiation values for vector-based methods.

1 | Introduction

The pursuit of mitigating our carbon footprint has prompted a rise in the utilization of renewable energy sources. Solar energy has experienced substantial growth in the 21st century, with an astounding increase of over 85.000% compared to the production in the year 2000 (Ritchie, Roser, and Rosado 2022). Technological advancements in the solar energy sector, particularly in the development of solar thermal and photovoltaic (PV) panels, have played a significant role in driving this remarkable progress. Several approaches have been developed to improve the estimation of optimal locations for the installation of solar PV systems. By 2030, ~60% of the planet's population is projected to reside in urban areas (UN Population Division 2018). For that reason, urban regions have become a crucial focus of research. Extensive investigations have been conducted worldwide, primarily targeting buildings, their roof and wall surfaces, to identify suitable locations for urban solar energy generation by use of simulation (Gonçalves et al. 2021; Hassaan, Hassan, and Al-Dashti 2021; de Sá, Dezuo, and Ohf 2022).

However, recent publications about the comparison of PV systems simulation tools (Kumar 2022; Milosavljević, Kevkić, and Jovanović 2022; Buzra and Serdari 2023) focus either on the performance of the PV system or on the quality of their input data. Both factors support the correct identification of locations for solar panel installation according to the solar irradiation intensity and its spatial distribution. Therefore, geographic information system (GIS) tools have been utilized to address challenges pertaining to the placement of

PV systems in urban environments (Agugiaro et al. 2012; Ramirez Camargo, Luis, et al. 2015; Viana-Fons, González-Maciá, and Payá 2020).

Nevertheless, each GIS software package has its specific data requirements, implements different radiation models and produces several results in terms of spatial and temporal resolution and, crucially, accuracy. Furthermore, ignorance of this uncertainty may lead to erroneous decisions and misguided planning for energy supply based on GIS-based approaches.

The goal of this article is to perform qualitative and quantitative analyses of GIS-based software simulation tools for solar potential via solar irradiation on roofs. The scope is the evaluation of these tools in several terms, such as data requirements, overall usability, user-friendliness and accuracy.

1.1 | Related Work

Freitas et al. (2015) wrote a review paper on modeling solar potential in urban environments. This paper includes an analysis of various solar radiation methods and the characteristics of several simulation tools, along with an overview of urban-oriented 3D models. However, these models do not incorporate semantic 3D city models (3DCM). Additionally, the literature review in their paper does not cover an analysis of raster and vector formats for the same study area across different locations on the planet.

A total of seven simulation tools are evaluated in this research. We compare them in terms of input and output data, as well as their accuracy, based on local weather data as ground truth. The reasons for selecting these software tools include availability, as our main target was open source simulation tools; the second reson is input data, specifically for vector-based tools as they offer support to 3DCM either directly or through already available conversion tools; the final reason is the maturity of the simulation tools, since there is a literature showing use cases in which these simulation tools have been used for the solar component. The inclusion in our analysis of the commercial software packages ArcGIS Pro and Rhino, is justified by their widespread usage and their relevance within the domain of GIS and architecture, respectively.

The tools, listed in alphabetical order, are; ArcGIS Pro v3.1 (Esri 2024) CitySim v2023.06 (Mutani et al. 2018), GRASS GIS v7.8.7 (Hofierka, Suri, and Huld 2007), Ladybug v1.6.33 (Roudsari and Pak 2013), SAGA GIS v8.5.1 (Conrad 2010), SimStadt v0.10.0 snapshot 20,230,307 (Duminil et al. 2022), and Urban Multi-Scale Environmental Predictor v4.0.2 (UMEP) (Lindberg et al. 2023). In the case of Ladybug, it works as a plug-in for Rhino 3D/Grasshopper (Roudsari and Pak 2013).

From the selected simulation tools, ArcGIS Pro, GRASS GIS, and SAGA GIS use clear-sky solar radiation models. The remaining simulation tools allow the use of weather data (as ground truth) of the weather station located at the point of interest of each location. Similarly to Kausika and Van Sark (2021), we perform no calibration of the results because our purpose is to analyze the quality of the simulation results against the ground truth based on the physical and geographical potential at each location, as proposed by Izquierdo, Rodrigues, and Fueyo (2008). The remaining of this section presents, in alphabetical order, a short overview of the simulation tools, including use cases for each of them.

1.1.1 | ArcGIS Pro

ArcGIS Pro is a proprietary GIS software developed and maintained by Esri. It provides several geoprocessing tools, two relevant ones for this research, which are available inside its solar radiation toolset (Esri 2024): Area Solar Radiation and Points Solar Radiation. The former tool facilitates the computation of insolation levels across an entire region, while the latter focuses on calculating insolation at a given (x, y) position.

The ArcGIS Solar Radiation toolset has been fundamental in research to compute the solar potential at multiple locations. (Wong et al. 2016) perform their analysis using a 3 m spatial resolution digital surface model (DSM) to assess the PV potential in Hong Kong while incorporating cloud cover probability maps to overcome the lack of these data in the ArcGIS method. Zhu et al. (2020) use the Point Solar Radiation toolbox from ArcGIS Pro to compute the annual solar irradiation. They employ monthly cloud cover data from World Weather Online (2024) between 2015 and 2017 to overcome the lack of these data. Kausika and Van Sark (2021) use ArcGIS Solar Radiation Tools to compute the PV potential in the Netherlands. Their simulation results are calibrated using ground measurements from the KNMI stations and further validated using data from the weather station located in Cabauw, which is a measuring mast of 213 m in height. Kausika and Van Sark state that the default models from ArcGIS Pro lead to substantial discrepancies in solar insolation values compared to ground data. They express that a calibration process shall be performed at different time scales based on the scope of the analysis and the spatial resolution of the input data.

1.1.2 | Grass GIS

GRASS GIS is a platform-independent, open-source software to manage geospatial data in multiple formats. In our case, it offers the r.sun module to compute the solar irradiation (global, diffuse, and direct) for a given location (Neteler and Mitasova 2010).

GRASS GIS has been used in various studies to analyze solar potential in urban areas. Weyrer (2012) uses it to elaborate a conceptual model for roof surfaces in Baton Rouge, Louisiana. Weyrer uses a 5 m spatial resolution DSM. However, at available locations, the author manually collected slope and aspect data of building roofs to improve the input data for the computation. Garegnani et al. (2015) evaluate the potential of a method that considers physical parameters to describe the availability of renewable energy sources using GRASS GIS as their computational software tool, and their study area is in the Alpine region. Liang et al. (2020) develop an open-source software tool for the computation of solar irradiation in urban landscapes. Their implementation supports 3DCM or meshes that enrich GRASS

GIS's r.sun model, which is the core algorithm for their solar potential computations.

1.1.3 | System for Automated Geoscientific Analyses

The System for Automated Geoscientific Analyses (SAGA GIS) is a multi-platform open-source software for geoscientific research and analysis. It supports multiple data formats and provides several geoprocessing tools for modeling and analyzing geospatial data. The Potential Incoming Solar Radiation Tool performs the calculation of incoming solar radiation (Conrad 2010).

Jochem, Wichmann, and Höfle 2009 use SAGA GIS to assess a 7 km × 7 km area in the city of Feldkirch, Austria. They use airborne LiDAR in their research for the creation of the urban scene as a raster, which is the basic data for the whole computational process. To calibrate the simulation results, they calculate the clear-sky index (CSI) using data from a nearby ground station. CSI is defined as the ratio of measured to modeled insolation values. However, the study does not include an accurate assessment of the solar irradiation computations. Gulben et al. (2019) use SAGA GIS to produce insolation maps in Iligan City in the Philippines, consuming LiDAR point clouds as a data source to create the urban scene raster. Similar to Jochem et al., the authors in this research also computed CSI values to calibrate the results. Ground data come from a pyranometer sensor placed on the roof of a building inside the study area. The highest discrepancies happen in July, which might be related to the sun's shifting orientation and the variations in the CSI maps for the observation period. Mujić and Karabegović (2023) present a comparative study using PVGIS, GRASS GIS, and SAGA GIS to evaluate the solar potential in Sarajevo City, Bosnia and Herzegovina. Although results show average annual solar potential values, further analysis is required since the spatial resolution of the computation methods varies. GRASS GIS and SAGA GIS use 30 m × 30 m DSM, while PVGIS uses 5 km × 5 km satellite imagery.

1.1.4 | Urban Multi-Scale Environmental Predictor

UMEP is an open-source climate service tool presented as a QGIS plug-in that offers tools for outdoor thermal comfort and climate change mitigation, among others. Its solar radiation tool, solar energy on building envelopes (SEBE), computes pixel-wise potential solar energy based on raster files.

Prieto, Izkara, and Usobiaga (2019) present a method for computing the solar potential of building roofs in an urban region based on LiDAR data; they use UMEP as the software tool to perform the solar potential computations. The article does not provide information regarding the accuracy assessment of the results. Polo and García (2023) evaluate the impact of different methods for the creation of a DSM with the scope of using them as the basis for raster-based solar potential computation, using UMEP to perform the solar potential analyses. The input data for this research include Google imagery and aerial LiDAR point clouds to create the urban scene raster. The results are evaluated using PVGIS data. However, the authors do not discuss in detail the discrepancies between the computed values and the ground data. Instead, their analysis focuses on locations where the discrepancies are large or small.

1.1.5 | CitySim

CitySim is an energy simulation software that is able to analyze the energy behavior of buildings by considering the environment in which they are located. It was initially developed as a command-line solver by Coccolo and Kämpf (2015). Years later, the energy consulting company Kaemco developed a graphical user interface (GUI; CitySim Pro) to facilitate the interaction of users with the solver (Mutani et al. 2018).

Using CitySim as the simulation tool, Mohajeri et al. (2016) evaluate the compactness of urban areas for solar potential in the city of Geneva, Switzerland. The hourly solar irradiation of each building surface is computed. Their research identifies that increasing compactness in a neighborhood leads to a decrease in annual solar irradiation (e.g., from 816 to 591 $\mathrm{kWh/m^2}$). Their findings suggest that urban compactness has a more significant impact on building-integrated photovoltaics (BiPV) for facades (decreasing from 20% to 3%) compared to roofs (decreasing from 94% to 79%). Chen, Rong, and Zhang (2020) use CitySim to calculate the wall temperature induced by solar radiation in an array of buildings, which were set as thermal boundary conditions in computational fluid dynamics (CFD) simulations. The relevance of their research lies in demonstrating that solar-induced wall temperature could be approximated by a uniform wall temperature based either on parallel or oblique wind conditions. Bensehla, Lazri, and Brito (2021) utilizes CitySim to compute the solar potential of various urban forms in the city of Constantine in Algeria. Several urban configurations are used in CitySim to correctly model the impact of the rooftop PV systems in building rooftops. In their findings, authors point out the absence of weather data measures for the computations, relying instead on typical meteorological yearly data provided by Meteonorm.

1.1.6 | Ladybug

The Ladybug Tools are a collection of free and open-source applications for environmental design, which allows view analyses, solar radiation studies, and sunlight-hours modeling, among others.

Groenewolt et al. (2016) use Ladybug in their design process to analyze the solar potential of installing PV modules on irregularly curved surfaces. Based on their results, they determined the characteristics (size and location) of the modules to be installed on the roof of the designed building. Freitas et al. (2020) evaluate the feasibility of using Ladybug to assess BiPV envelopes in several office buildings in Brasília, Brazil. Their method considers measured data of end-use energy consumption and urban and building morphology of the study area. Their research is focused more on the architectural analysis of the PV panel installations rather than the accuracy of their computations, with no information about the latter in their article. Rostami, Nasrollahi, and Khodakarami (2024) assess the daylight availability and energy consumption in urban canyons and blocks in Ilam City, Iran. Additionally, they explore the correlation between the urban morphological characteristics of urban blocks and key indicators defined for the research using Ladybug Tools. The 3D representation of the urban area includes city objects such as buildings, trees, and reliefs. However, these elements are represented at lower levels of detail; buildings are modeled without elements such as windows or balconies, and in the case of trees, although their height values were obtained via on-site measurements, they are modeled using simplified geometries.

1.1.7 | SimStadt

SimStadt is a Java-based, multi-platform, open-source software tool that performs several analyses of buildings on an urban scale. It offers energy-related applications such as solar energy analysis, PV potential, and energy demand computation based on the energy-balance method. User interaction is done by means of a GUI, although its functionalities can also be accessed through a command shell.

Rodríguez, Nouvel, et al. (2017) determines the solar PV potential in urban regions using 3DCM of Ludwigsburg County, Germany, with SimStadt as the core component for the computations. They explore two scenarios using different PV modules. The results show that it is possible to achieve high annual rates of electricity demand in several municipalities in the considered scenarios. Rodríguez, Duminil, et al. (2017) present different approaches for computing urban solar irradiation in large areas. Their proposal includes three methods: the Perez Sky model, the Simplified Radiosity Algorithm, and a tiling method developed by the authors. All methods are implemented using SimStadt. Authors quantify the influence of shading using the concept of the Urban Shading Ratio. In highly dense urban areas, this index may reach 60% for facades and 25% for roofs. Würstle et al. (2020) present a concept for an urban energy dashboard built using Open Geospatial Consortium (OGC) standards, using 3DCM and SimStadt as the energy simulation tool. The study regions include Landkreis Ludwigsburg, Germany; Brooklyn, New York City, USA; and Wüstenrot, Germany.

2 | Materials and Methods

2.1 | Method

This section presents the method developed and followed in this research and gives a general introduction to each of the steps of the workflow shown in Figure 1. The method is designed to analyze and compare the GIS software tools as standardized as possible despite their different characteristics, data requirements, etc. The workflow is split into four main parts, which are explained in the following sections of this paper. Please note that, in Figure 1, the text in gray refers to the respective section that further describes the step.

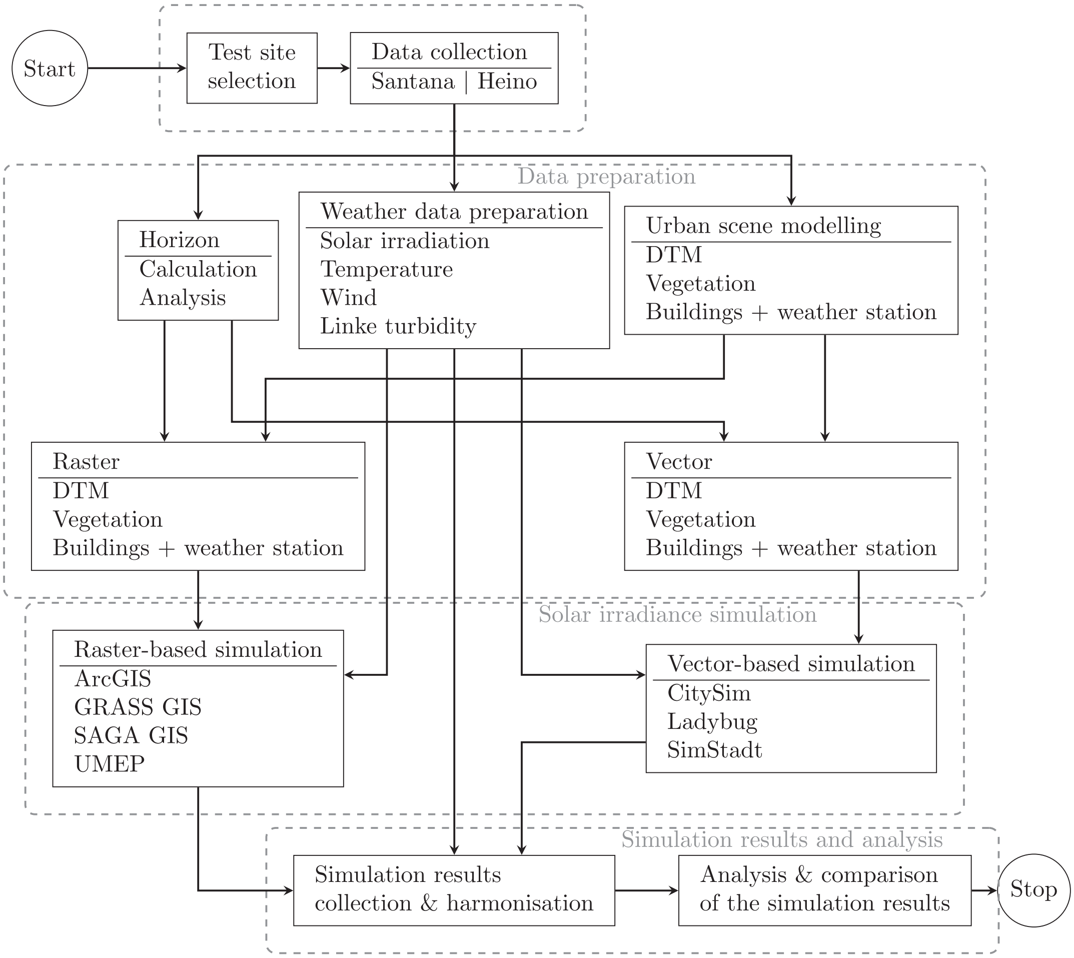


FIGURE 1 | General workflow of the comparative method described in this article.

The first part of the workflow corresponds to the *Test site selection* and the associated *Data collection*.

The Data preparation step deals with the required processes to create the input data for the simulation tools, and it is split into three parts based on the purpose of the data: *The Urban scene modeling* block corresponds to the geospatial data that are required for the solar irradiation simulation. *The Weather data preparation* block corresponds to the weather data required as input for some of the simulation tools as well as for the accuracy assessment of the simulation results. The *Horizon* block deals with the calculation of the horizon mask at the study area, as well as the delimitation of the study area. In the *Raster* and *Vector* blocks, the respective files for the simulation tools are created according to the required data type, taking into consideration the terrain model in terms of a DSM, which contains all features above the surface. A DSM represents the urban areas better than a digital terrain model (DTM) since they include vegetation, buildings and, specifically to our research, also the weather station.

The Solar irradiation simulation step corresponds to the execution of the selected software tools.

The Simulation results and analysis step entails two blocks: the *Simulation results collection & harmonization* block collects and structures the output of the simulation tools and consolidates the values so that they are comparable between the software tools as well as the ground truth; the *Analysis & comparison of the simulation results* block performs a quantitative analysis of the results.

2.2 | Test Site and Data Collection

We apply our method to two study areas, São Paulo in Brazil (Figure 2) and Raalte in the Netherlands (Figure 3), to reduce

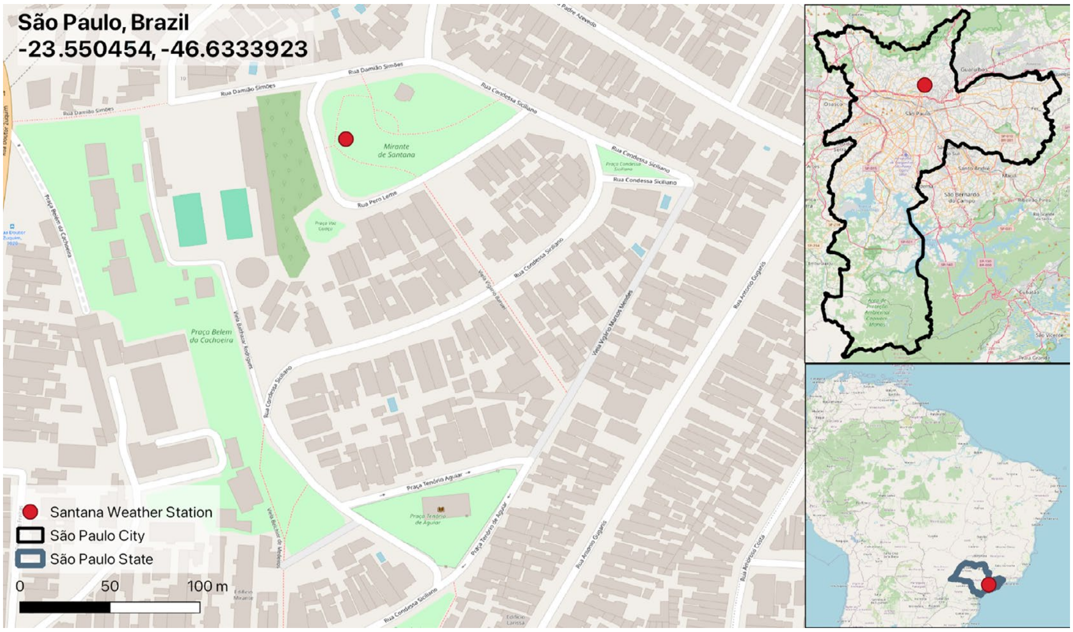


FIGURE 2 | Santana District in São Paulo, Brazil. Location of the study area.

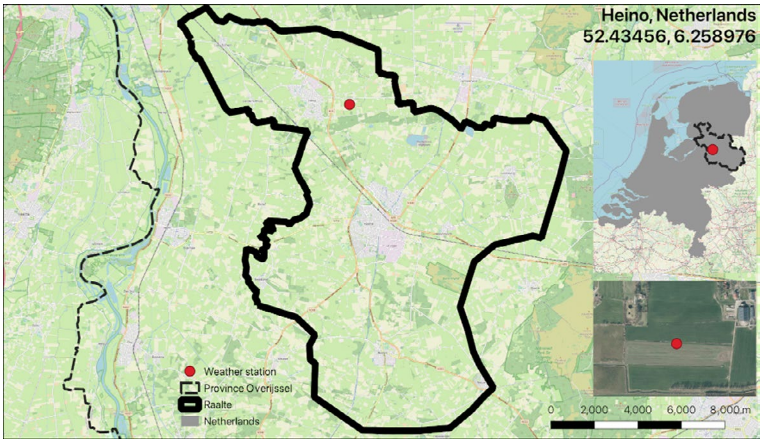


FIGURE 3 | Heino village in Raalte, the Netherlands. Location of the study area.

the impact of local effects and ensure broader applicability. These sites have been chosen because they represent opposite conditions. These contrasting locations allow the replicability of our method in other regions with similar characteristics as chosen study areas. Additionally, it allows the comparison of regions with different characteristics.

São Paulo is a densely populated urban centre located in the southern hemisphere of the Tropic of Capricorn. It is characterized by minimal vegetation, significant elevation changes, several unstructured urban developments, and several informal settlement areas named favelas (Portuguese). This location represents metropolitan regions.

On the other hand, Raalte is situated in a mostly rural part of the Netherlands, which has a flat landscape with few buildings and abundant vegetation. This region represents greenery in unpopulated rural areas.

We applied the same method for data collection and integration in both locations. That includes the collection of spatial and non-spatial datasets. From these datasets, we created both vector- and raster-based datasets as input for the simulations, ensuring comparability across the two diverse regions. Table 1 provides a summary of the data used for this research.

2.2.1 | Santana

"Mirante de Santana" is a weather station in the northern part of the city of São Paulo. Its surroundings were chosen as the area of the first study case, as it allows the comparison of measured data and simulation values at the same position. For the Santana test area, two main data sources were used in this research:

GeoSampa, which is a public open geodata portal from the municipality of São Paulo (GeoSampa 2023), provides the following data sets: footprints of buildings with the following attributes: ID, area, and height; a LAZ-encoded point cloud with ASPRS15 LiDAR class values from which we produce the raster-based DSM.

The database of the National Institute of Meteorology (INMET) provides weather data. We use the data from two weather stations, "A701 São Paulo Mirante" and "83781 Mirante de Santana"; both stations lie side by side. We are required to use the data from two weather stations because Mirante de Santana does not collect cloud nebulosity values. Further details about the data preparation and creation of the required files are described in (Giannelli 2021).

2.2.2 | Heino

For this study area, several height sources at different spatial resolutions were used. They are the Dutch elevation database (AHN3) consisting of point clouds and DSMs at 50 cm, 5 m, 25 m grid resolution (Stuurgroep AHN 2019). For the vector data, we used the open 3D building data set of the Netherlands (3DBAG) (Peters et al. 2022).

In the case of the weather data, we use the available local weather data from the Royal Netherlands Meteorological Institute (KNMI), which offers the meteorological data collected from 48 weather stations in the country (KNMI 2024).

2.3 | Data Preparation

An important objective of this work consisted of preparing both geospatial and weather input data in the most similar way for the GIS software used in order to minimize potential discrepancies in the simulation results. Wherever edits or changes to the raw input data were made, they were replicated in all formats so that the produced input data could be considered equivalent but "just" in a different data format. One example is the weather station location: the pixel height value of the raster DSM was modified (in our case, raised) to the corresponding height value of the weather station. The reason is that we wanted to model the 3D position of the weather sensor both in the vector and raster datasets. For the same reason, in the vector-based data sets, we added an "artificial" building at the location of the weather station with a planar roof of 1 m^2 corresponding to the position of the sensor. Table 2 summarizes the data requirements for all simulation tools; "O" stands for optional, and "M" stands for mandatory. If no information is provided, this means that the software tool does not deal with the corresponding input specifically.

For the vector-based simulation tools, we use a DTM to represent the earth's relief, and we incorporate features such as buildings and trees by means of additional datasets. However, these elements are directly included in the raster-based DSM,

TABLE 1 | Summary of the input data sets in our research.

	São Paulo, Brazil	Raalte, the Netherlands
Geospatial	GeoSampa: • 2D building footprints • Aerial point cloud • Raster-based DSM (1 m)	DBAG: 3D Buildings AHN3: Aerial point cloud Raster-based DSM: • AHN3: 0.5 m, 5 m
Weather	• A701 São Paulo Mirante: Total sky cover (Nebulosity) • 83,781 Mirante de Santana: Global, direct, diffuse, solar irradiation, dry-bulb and dew-point temperature, station pressure, relative humidity, wind speed and direction	• KNMI Heino weather station: Typical yearly values of global, direct, diffuse solar irradiation, ambient and ground temperature, wind speed, could coverage, pressure, rainfall

TABLE 2 | General data requirements for each simulation tool.

	Raster-based				Vector-based		
Item	**ArcGIS Pro**	**GRASS GIS**	**SAGA GIS**	**UMEP**	**CitySim**	**Ladybug**	**SimStadt**
Horizon		O	O		O		
Relief	DSM	DSM	DSM	DSM	DTM	DTM	
3D buildings					M	M	M
Vegetation				O	O	O	
Weather		O	O	M	M	M	M

TABLE 3 | Data sets resulting from the Data preparation step with the indication of which data are used in each simulation tool.

	Raster-based				Vector-based		
Item	**ArcGIS Pro**	**GRASS GIS**	**SAGA GIS**	**UMEP**	**CitySim**	**Ladybug**	**SimStadt**
Horizon		X	X		X		
Relief (DSM)	X	X	X	X			
Relief (DTM)					X	X	
D Buildings				X	X	X	X
Trees				X	X	X	
Solar irradiation values				X	X	X	X
Dry-bulb and dew-point temperatures				X	X		X
Station pressure				X	X		X
Relative humidity				X	X		X
Wind speed				X	X		X
Total Sky cover				X	X		X
Linke turbidity		X	X				

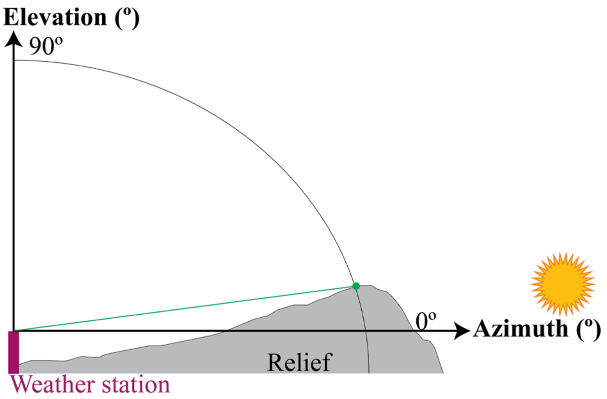


FIGURE 4 | Sketch of horizon mask for a given direction. The green line indicates the lowest height above the Horizon of the sun in that location and direction.

which captures both the relief and any objects on it. At the end of the section, Table 3 shows a summary of the resulting data sets created for this research and their usability by the simulation tools.

2.3.1 | Horizon

The Horizon analysis corresponds to the skyline computation from a given position in all horizontal directions (Calcabrini et al. 2019). For the given position, the results consist of a so-called horizon mask, i.e. a set of elevation angles associated with their respective azimuth values. Each elevation angle indicates the minimum height over the Horizon at which the sun is visible in that direction (Figure 4). The horizon analysis allows us to speed up solar simulation time. No simulation is needed if the sun is below that elevation angle for a specific azimuth, as it is hidden by some feature (e.g., a mountain).

The first step in calculating the Horizon is the generation of a (or improvement of an existing) raster DSM covering the study area, with its centre at the location of the specific point of interest. If an existing DSM is available, data gaps or holes, which represent missing height values (Figure 5), must be filled, for example, by means of bi-linear interpolation. The final step involves modifying the height value at the point of interest, setting it to the precise height of the weather station (rather than the underlying terrain, as in our case).

The pre-processed DSM is used as input data for the Horizon calculation. We compute the azimuth, distance, and angle of elevation from all DSM pixels against the location of the point of interest. A lower-resolution DSM covering more extensive areas is needed to check whether far-away elements may act as an obstacle to the point of interest. If that is the case, they should be included in the horizon profile that will be used in the following processes.

In our approach, the higher the spatial resolution of the input DSM, the smaller the step for the horizon analysis. Based on this, for very-high-resolution rasters ($\leq 1\,\mathrm{m}$), we used intervals of 100 m; for 5 m resolution DSM, we used a 1 km interval; and, finally, for the 25 m raster DSM, we used a 4 km interval.

The main result of this process is the definition of the size of the study area based on the main input geospatial data set, which is critical to consider as the following parts of the workflow are computationally expensive.

2.4 | Weather Data

In the case of Santana, we downloaded the required weather data from INMET data sets containing records from 26/07/2006 until 25/07/2021. We averaged the solar irradiation values for each hour from 15 years of historical observations in order to obtain the hourly values of a typical year.

CitySim, SimStadt, and UMEP require diffuse values in addition to the global and direct ones as available in the local weather data set. Figure 6 shows the daily solar global irradiation values of the input weather data set for Santana. However, this data set does not provide values for diffuse radiation. Therefore, we use CitySim as a weather data pre-processor to generate the diffuse and direct beam values for all simulations. This tool implements the (Direct Insolation and Simulation Code) DISC-model for its calculations (Maxwell 1987).

The KNMI weather stations measure air temperature, humidity, air pressure, wind speed and direction, the amount of precipitation, the type of precipitation (rain, snow), the altitude and amount of cloud, solar radiation, and horizontal visibility. One of these stations is situated in Heino. Figure 7 shows the daily solar global irradiation values of the input weather data set for Heino. For areas lacking accessible data, alternative sources, such as the OneBuilding project, are available (Lawrie and Crawley 2023). The data availability on this platform facilitates the replicability of our experiment in regions where no local weather data are available.

The Linke turbidity factor is another weather parameter requested by GRASS GIS and SAGA GIS. Therefore, we decided to use the data available at SoDa (2010), which is a raster-based data set of monthly values with a very low spatial resolution of $1/2^{\circ} \approx 0.08333^{\circ}$ in both latitude and longitude. Additionally, the Linke values are "multiplied by 20 for storage constraints" (SoDa 2010). Three steps of pre-processing were needed to use the soda data set. First, the reprojection from the source coordinate reference system to the corresponding one at the test area location. Second, the Soda raster files were re-sampled to the exact spatial resolution as the input DSM. Third, the pixel values were divided by 20. Since the extent of the study areas in

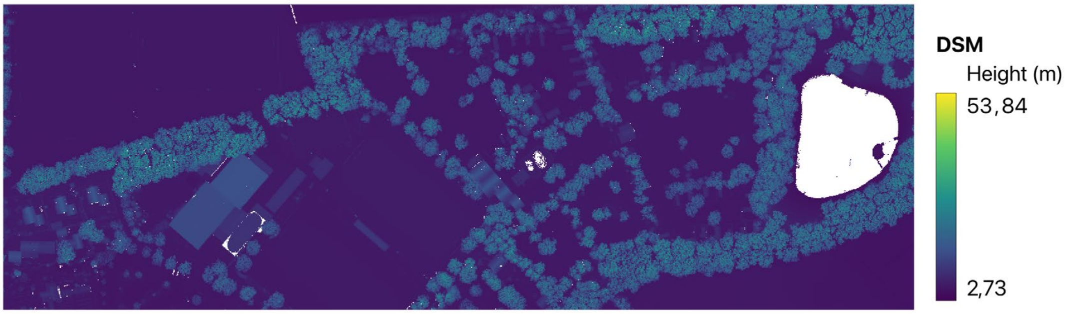


FIGURE 5 | Excerpt of the DSM around Heino, white color areas correspond to no data, in this sample all data gaps correspond to water bodies.

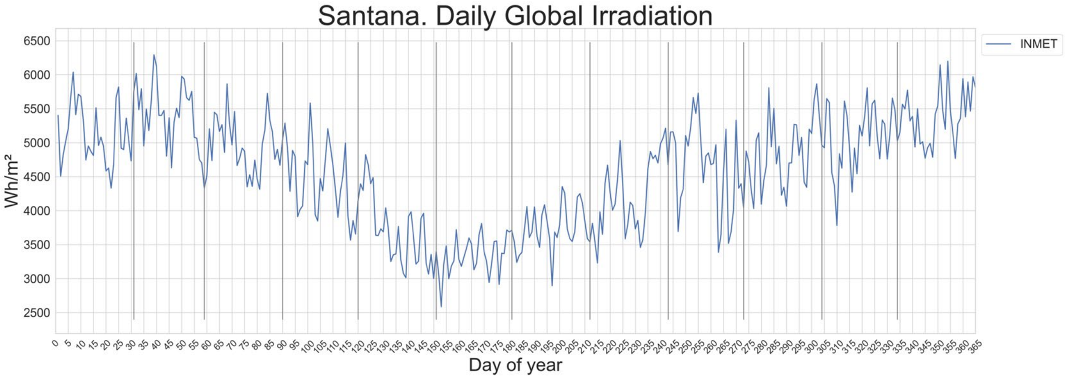


FIGURE 6 | Global solar irradiation at the weather station in Santana.

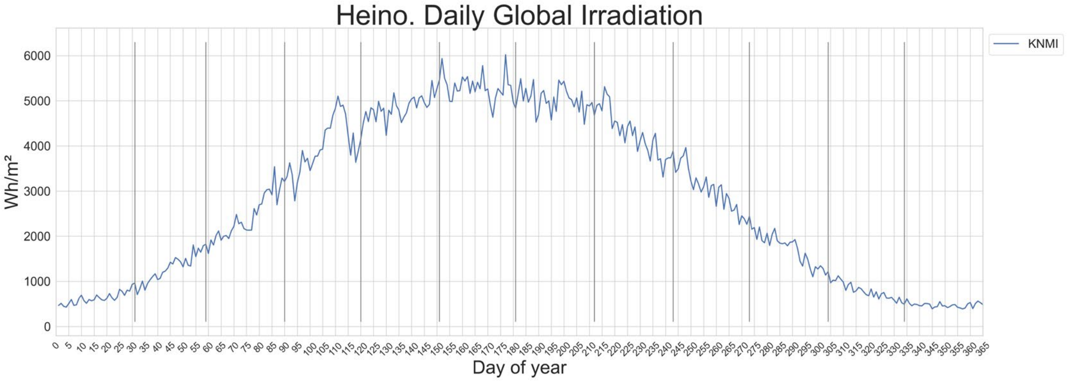


FIGURE 7 | Global solar irradiation at the weather station in Heino.

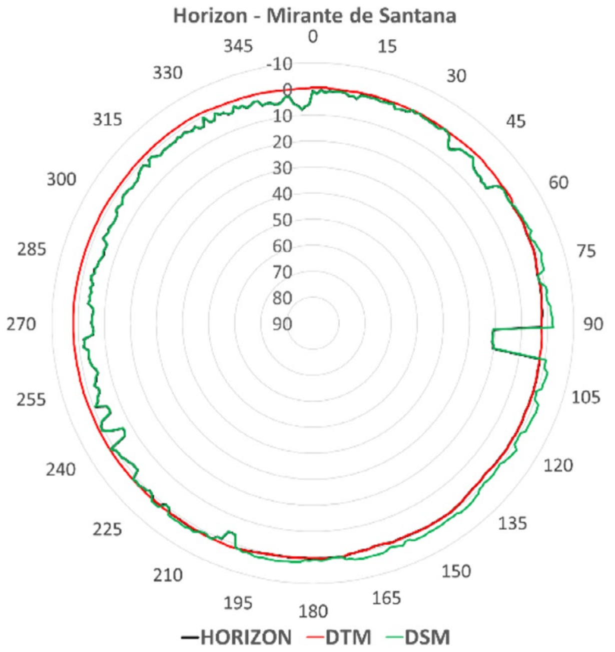


FIGURE 8 | Santana, horizon view at the weather station using the DTM and DSM, azimuth, and elevation angles in decimal degrees.

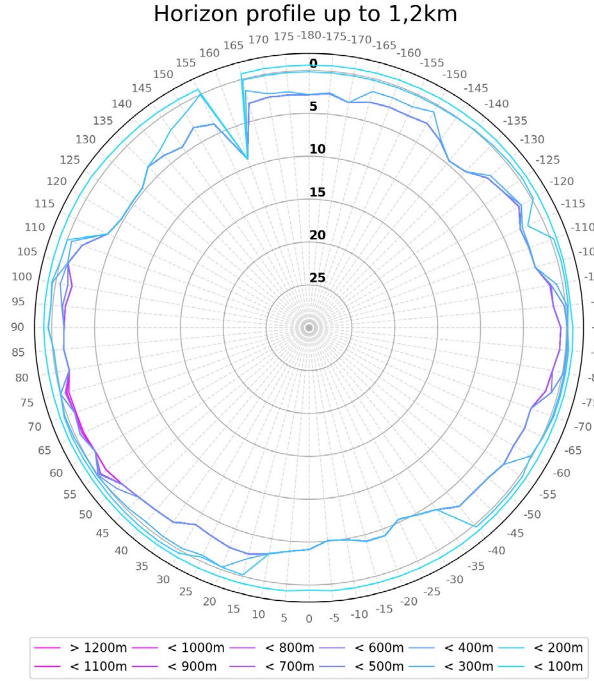


FIGURE 9 | Heino, Horizon view at the weather station using the AHN3 50 cm DSM. Azimuth and elevation angles in decimal degrees.

our implementation is relatively small (a couple of kilometers per side), most of the re-sampled SoDa raster files contained just one value.

Although this article does not include graphs of the simulation results without Linke Turbidity data, incorporating these data does influence the computation of the solar potential. Both GRASS GIS and SAGA GIS employ clear-sky models for their computations. For this reason, usage of the Linke Turbidity parameter is highly suggested by GRASS GIS developers (Hofierka, Suri, and Huld 2007).

2.5 | Urban Scene Modeling

As previously mentioned, our goal is to produce as similar as possible geospatial data sets that contain the features representing the urban scene in the study areas. To do so, we include terrain, vegetation, and buildings. We use FME and Python for the 3D modeling and data format conversion.

2.6 | Raster

For ArcGIS Pro, GRASS GIS, and SAGA GIS, the minimum input data is a DSM of the area of interest, which extension results from the Horizon step of the workflow in Figure 1. Here, we discuss the specifics for each of the study areas that differ from the previously created DSM.

In Santana, the raster that covers the study area is a 1 m grid resolution DSM and spans over an area of circa 2.9 km × 2.7 km around the weather station. We created an additional relief raster, which has a more extensive coverage than the study area to account for potential occlusions resulting from nearby objects and topographical features. It is a 50 m spatial resolution DTM that covers a region of 30 km × 20 km, predominantly to encompass the topographic features (hills) situated to the north of the weather station. Figure 8 shows the result of the horizon calculation for Santana. Please note that the figure contains three skylines, although only two of them are visible. The Horizon follows the same behavior as the DSM.

In the case of Heino, the extent of the raster file was determined by the results of the horizon analysis. Figure 9 shows the result of the sky view calculation at the Heino weather station using the AHN3 DSM at 50 cm. The highest obstacle is less than 200 m away with an azimuth of 156° and an elevation angle of 9°. As we do in Santana, we use additional relief rasters that have a more extensive coverage than the study area. However, due to the constant flat relief of The Netherlands and surroundings, we perform the analyses at different spatial resolutions and different distances from the study area. We split the results into multiple images due to the overlap between the plot lines.

Figure 10 shows the resulting horizon view using the lower resolution DSMs; from left to right, the spatial resolution of the input DSM decreases. Figures 9 and 10 use the same color palette going from cyan (closest) to magenta (furthest). Figure 10a,b show the relief as far as 20 km and as far as 40 km does not influence the horizon using the AHN at 5 and 25 m resolution, respectively. Although Figure 9 should show 13 lines each, this is not the case because Horizon has hardly any variation for the study area. Our plotting method was designed to add the horizon line from the furthest to the closest object, yet the variation is minimal for both input datasets. Therefore, only blue to cyan colors are visible.

We remove from the analyses of the lower resolution DSM the data of the area covered by the very-high-resolution DSM since our objective is the analysis of the impact of far-distance obstacles at the location of the weather station rather than an accuracy assessment of the different data sets in the overlapping area. Figure 10 indicates that the weather station is not affected by far-away obstacles. Since Figure 9 shows that there is no relevant change in the Horizon from 1.200 m away from the weather station, we decide to use for our analyses a square with a side of 2.400 m as the study area.

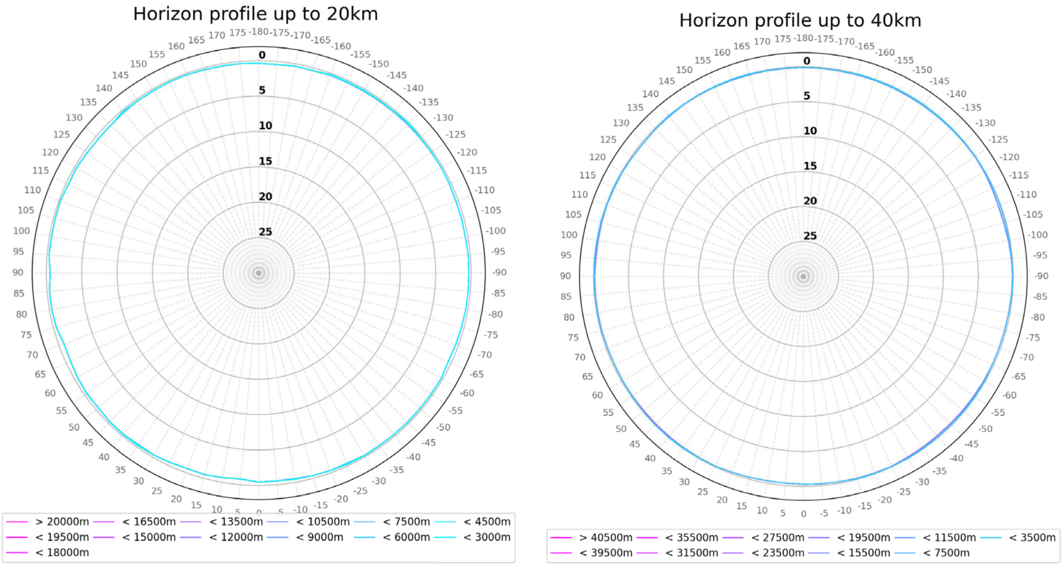
2.7 | Vector

In both study areas, we use the LiDAR points classified as ground for the relief. The process begins with the generation of a TIN. The building footprints from the 3DCMs are then used to clip the TIN areas beneath buildings. This step's scope is to reduce the computation of the simulation time (by reducing the number of geometries) and to avoid errors since these surfaces do not receive sunlight. We perform this task using FME Form (Workbench) v.2023.1 (Safe Software 2023). Additionally, we include an artificial building with a footprint of 1 m×1 m at the position of the weather station, with a horizontal roof surface adjusted to correspond to the height of the station. We prepare a standard vector-based input data set using CityGML (Gröger et al. 2012).

2.7.1 | Santana

We reconstruct the 3D building models using their 2D footprints as, unlike in the Dutch case study, there is no readily available 3D city model. For the height at ground level, the lowest height value is selected of the vertices that intersect the DTM. For the building height, we filter the point cloud, keeping only those that are classified as buildings and extract the mean height value. We generate prismatic geometries by extruding the footprints with a ≥ 3 m height constraint, so we keep as buildings those that have liveable height value. For our experiments, all buildings are considered shadow-casting objects. Additionally, flat roofs are the most frequent roof type in Brazil. Therefore, this representation suits our needs. Building geometries were classified as Level of Detail (LoD)2 surfaces (Gröger et al. 2012, 64), with their corresponding Roof, Wall, and Ground Surface semantic attributes.

The point cloud is also used for vegetation modeling. First, we filter the points classified as vegetation that do not overlap the



(a) $5m$ DSM

(b) $20m$ DSM

FIGURE 10 | Horizon view at the weather station at Heino using AHN3 at 5 m (a) and at 25 m (b) resolution. Azimuth and elevation angles in decimal degrees.

building footprints. Although, commonly, high trees overlap with constructions, we decided not to do that to avoid having 3D features intersecting each other. The location of the geometry of the trees is determined by a 5_{m} clustering to split vegetation areas into single trees; smaller areas are modeled directly without additional pre-processing. The tree crowns are calculated from the resulting clusters using a 3D convex hull. For the trunks, we model a square-shaped (fewer geometries for the upcoming simulations) 25 cm side extruded upwards from the DTM to the lower part of the computed crowns. Results are classified and stored as CityGML *SolitaryVegetationObject*. Figure 11 shows an excerpt of the 3D reconstruction workflow, Figure 12 presents the 3D city model of Santana. More details can be found in (Giannelli 2021).

2.7.2 | Heino

Buildings in the study area are extracted from the open 3D building data set of the Netherlands (3DBAG) (Peters et al. 2022). We follow a different approach to model the vegetation and represent trees as implicit geometries. Therefore, geometries contain fewer surfaces, leading to lower computational time for the simulation tools. The vegetation is used as shadowing objects, and this method is, therefore, sufficient. First, we filter from the point cloud those that are classified as vegetation; then, we aggregate them by a 5 m clustering. The resulting areas are then used as the input for a fishnet grid of 5 m to locate each tree inside the area. The height value is computed as the height difference between the TIN created for the relief and the point cloud at the given location.

For the relief, we use the raster DTM at 0.5 m available from (Stuurgroep AHN 2019). The workflow followed is the same as Santana's. Figure 13 shows the CityGML-based model of the study area and a close-up view of the "artificial" building at the weather station location.

3 | Solar Simulation

This section presents the data prerequisites and limitations for each simulation tool. We provide a detailed overview of the radiation models implemented in the chosen solar simulation tools is beyond the scope of this article. Instead, we provide an overview of the data requirements, setup, and parameters for each tool, along with references to research work where these tools have been applied. All simulation tools share fundamental prerequisites, including atmospheric conditions, surface characteristics, and location data.

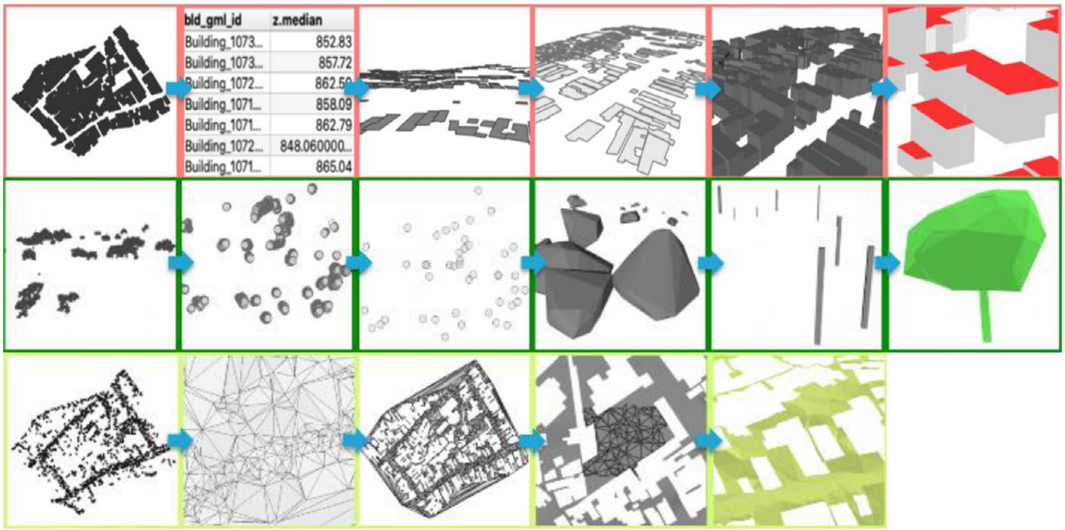


FIGURE 11 | Santana 3D modeling workflow.

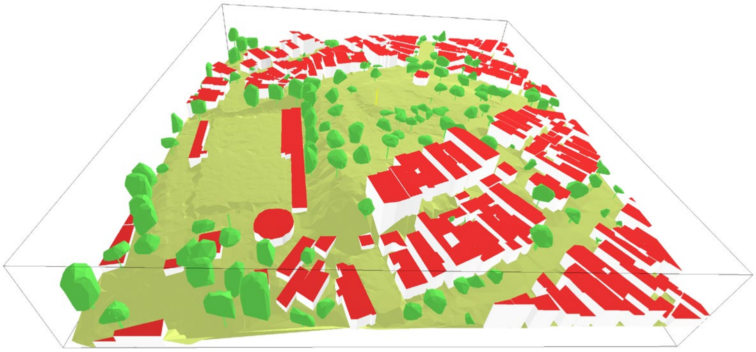
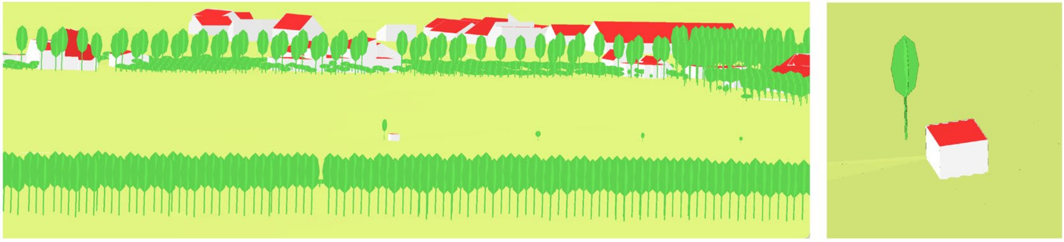


FIGURE 12 | Santana, example of the vector-based modeling of the urban scene.



(a) CityGML file

(b) Weather station

FIGURE 13 | Heino, excerpt of the created CityGML file centring the artificial building representing the weather station.

Nevertheless, the selected software tools are not treated as complete black boxes. If possible, causes of potential discrepancies in the results are considered. For example, the raster-based tools ArcGIS Pro, GRASS GIS, and SAGA GIS all perform their computations using clear-sky conditions. However, GRASS GIS and SAGA GIS offer the option to add additional data, such as the Linke Turbidity. Additionally, all simulation tools compute the slope and aspect from the input DSM file. ArcGIS Pro and GRASS GIS also allow the exporting of generated slope and aspect raster files. For the sake of completeness, by means of map algebra, we have performed a test to check for differences between the slope and aspect maps of the two software tools. As no significant differences have been found (the absolute values of the differences vary from $7.62939e^{-6}$ to 0), this means that, for our research, slope and aspect maps generated by ArcGIS Pro and GRASS GIS are equivalent.

3.1 | Raster-Based Tools

3.1.1 | ArcGIS Pro

When it comes to ArcGIS Pro, we used the Area Solar Radiation tool in this study. It requires only a raster-based terrain model, either a DSM or DTM. Other parameters can be extracted from the input raster by the software or manually tuned by the user, like the sky view, slope and aspect input type, the number of calculation directions of the sky, and the zenith and azimuth divisions. In the case of the slope and aspect maps, the user can indicate if they should be included in the calculation or if the simulation should be performed on a flat surface; if slope and aspect parameters are chosen, they are calculated once at the beginning of each simulation run, no matter which temporal resolution is chosen. The tools only allow the input of the terrain model as a data source, whereas all other parameters are non-data set variables. Therefore, even though the slope and aspect data sets are fixed and could be prepared in advance for reuse, these two are repetitively calculated every time the tool is run for each day of the year.

For each test area, a Python script was developed to run the simulation tool for all 365 days of the year, using a time interval of 1 h. Although the tool requires a date interval from the user, it only uses the input data to identify whether it is a leap year or not. The output consists of several raster files in which pixel values are expressed in Wh/d/m^2: global and direct irradiation rasters. Additional options, such as diffuse irradiation and direct duration, are also available.

3.1.2 | Grass GIS

When it comes to GRASS GIS, r.sun is the module used by this software. Prior to using it, users are suggested to pre-compute slope, aspect, and horizon maps using additional modules provided within the software to speed up the simulation process. The module supports atmospheric parameters, such as the Linke atmospheric turbidity coefficients, albedo, real-sky beam, and diffuse radiation coefficients. For the Linke values, we used the rasters created during the weather data preparation, as described in Section 2.4.

Python scripts were written to automatise the execution of the simulations. The output consists of several raster files per simulation day, which depict the daily solar irradiation values with a time interval of 1 h. Additional, r.sun is a highly demanding computational tool. Therefore, the documentation (Hofierka, Suri, and Huld 2007) points out possible out-of-memory problems—which we faced in our experiments. For that reason, we used a horizon step of 3°, which is the lowest value we could use without encountering the above-mentioned out-of-memory issues.

3.1.3 | SAGA GIS

SAGA GIS only requires a DSM as a mandatory input data set. Other input datasets, such as the Sky View Factor raster (computed using SAGA's tool from the DSM file) and the Linke turbidity coefficient, are optional. However, they were included in our experiments.

The actual simulation runs were automatised by means of shell scripts, one for each day of the year. As we did with ArcGIS Pro and GRASS GIS, we set up the simulation time step to 1 h. For the atmospheric effects, the Linke turbidity coefficient rasters, previously prepared and described in Section 2.4, were used. The output consists of raster files containing direct, diffuse, and total insolation. The latter is, however, optional as it is the sum of the former two.

3.1.4 | Urban Multi-Scale Environmental Predictor

UMEP has slightly different data requirements than the previous raster-based GIS tools. The mandatory input is a DSM that contains ground and building features. Other required rasters are the wall height and the wall aspect. These can be computed from the input DSM using their pre-processor named "Urban Geometry: Wall Height and Aspect" (Lindberg et al. 2023).

The tool supports additional input rasters, such as the Canopy Digital Surface Model (CDSM) and Trunk Zone Digital Surface Model (TDSM). UMEP offers the Spatial Data: Tree Generator pre-processor to compute them. We created our CDSM from point clouds, following their principle: pixels with no vegetation should be zero, and their value corresponds to the height at the corresponding location. In our experiments, regions with no vegetation were set to null-value pixels. We did not encounter any errors at run-time.

3.2 | Vector-Based Tools

3.2.1 | CitySim

In terms of data requirements, CitySim works with its data model, i.e., the CitySim XML file format (Coccolo and Kämpf 2015). However, there are two methods for data transformation into this format. The first one involves CitySim Pro, which offers a GUI, that supports the import of 3D files, i.e., CityGMl or DFX files. Further information is available at (Kaemco 2023). One major drawback of this import method is that only a set of default physical parameters is associated with the buildings, and the user must otherwise edit and change each one of them manually. The second method is a Python script that consumes a 3DCityDB instance and extracts the buildings, relief and vegetation data for a given study area, as indicated by the user (Yuzhen and León-Sánchez 2023). It automatically adds the corresponding building physics parameters, retrieving them from another database. Since we are interested in the solar irradiation values for the location of the point of interest (the weather station position), any other objects in the study area other than the artificial building are considered shadow-casting objects to reduce the simulation time. The software produces results with an hourly temporal resolution.

Mandatory data requirements are 3D geometry data and a tab-separated (TSV) climate file, and an optional horizon file can be utilized, which is the one created in sec:WeatherData. The climate file contains hourly values for cloud nebulosity, global horizontal irradiation, wind velocity, wind direction, air and surface temperature, total precipitation, and relative humidity for a representative year (Mutani et al. 2018).

3.2.2 | Ladybug

In order to prepare the urban scene according to the Ladybug's requirements, we triangulated the CityGML geometries by means of an FME workbench (Safe Software 2023) before converting them into a layer-based DWG file, as this data format is necessary for compatibility with Rhino Ladybug Tools LLC (2022). Each surface mesh is stored in a separate layer named according to its GMLID. Similar to CitySim, the artificial building representing the weather station is the only object for which the solar values are simulated and stored. In contrast, all remaining objects within the study area are categorized as simple shadow-casting objects. This approach helps to reduce the overall simulation time.

The weather data required for the simulation consists of hourly values of direct and diffuse irradiation specific to a designated Ladybug location, in our case, the basic metadata of the weather station: name, time zone, longitude, latitude and height. Subsequently, the irradiation values are transformed to conform with the sky matrix input requirements of the simulation tool. The output consists of a TSV file with the hourly values for each of the input surfaces; the order of the results corresponds to the input surfaces for simulation.

3.2.3 | SimStadt

SimStadt is capable of processing LoD1 or LoD2 CityGML buildings (Duminil et al. 2022). As for weather data, it currently supports three available methods: INSEL (Schumacher 2014) and PVGIS (EU Science Hub 2022) weather databases, as well as a TMY3 file (Wilcox and Marion 2008). For our experiments, we utilize the TMY3 file, which requires the inclusion of the following mandatory parameters: direct normal irradiance, global horizontal irradiance, diffuse horizontal irradiance, dry-bulb temperature, total sky cover, dew-point temperature, station pressure, relative humidity, wind speed, and wind direction.

In terms of radiation modeling for our simulations, we chose the one described by Perez et al. (1987), as it is the sole available in the software that incorporates the shadowing effect of nearby buildings. The results consist of average yearly irradiation values per surface, stored in a CSV file. To obtain hourly values, the user must enable the results cache storing option from the GUI and explore the cache folder in the root directory of the project. Although the cache contains the results per surface, they do not contain the surface's ID, which challenges the identification of the results for the corresponding RoofSurface.

4 | Results

4.1 | Simulation Results and Quantitative Analysis

Simulation results were consolidated and aggregated into daily, monthly, and yearly values to proceed with the quantitative analysis.

4.1.1 | Solar Irradiation Result Comparison

4.1.1.1 | Daily values. Figure 14 shows the daily values resulting from the raster-based simulation tools for Santana. It is noteworthy to remember here that the raster-based simulation tools implement a clear-sky model, i.e. the results are indeed expected to differ from the weather station data used as ground truth. In general, simulated values should be higher than those of the weather station. Therefore, the ground truth data is added to Figure 14 to provide a sort of reference.

Looking at the results plotted in Figure 14, it is interesting to note that all simulation tools tend to follow the same pattern as the ground data, and, as anticipated, none of them produce values close to the measured ones. However, GRASS GIS values are surprisingly nearly always lower than the ground truth data, with the most significant difference being − 74.64% for day 31. SAGA GIS, instead, always provides higher values than the weather station, as was expected, with the maximum difference of 98.07% for day 153. The ArcGIS Pro output has closer values during the autumn and spring seasons, lower underestimation values in winter, and a particular behavior during summer.

The divergences between multiple software tools depend most likely on the implemented solar radiation model, which, however, especially in the case of ArcGIS Pro, cannot be further tuned as it acts as a sort of black box as the model's details are not published due to commercial purposes.

Figure 15 shows the daily values of the output of the vector-based simulation tools for Santana. All simulation tools follow the pattern of the ground data, which is expected since it is the data source for the computation. CitySim has an average underestimation of ~4.2%. In the case of Ladybug, the output has an average difference of ~2.2% against the ground truth. However, SimStadt (violet line) shows a considerable deviation from the ground truth values, on average ~24.7% on a yearly basis, with higher values during the winter season (~−47%).

Figures 16 and 17 show the raster and vector-based daily values of the output of the simulation tools for Heino, respectively.

For the raster-based results, the same consideration still applies as in the case of Heino. However, the significant difference is that GRASS GIS data values, unlike in Santana, are higher than the ground truth. In the case of ArcGIS Pro,

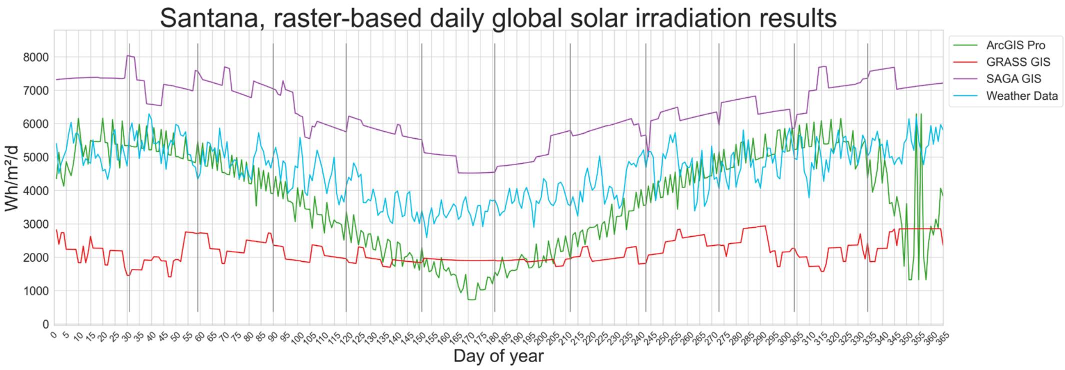


FIGURE 14 | Santana, time series of the raster-based simulation results and, for reference, the weather data.

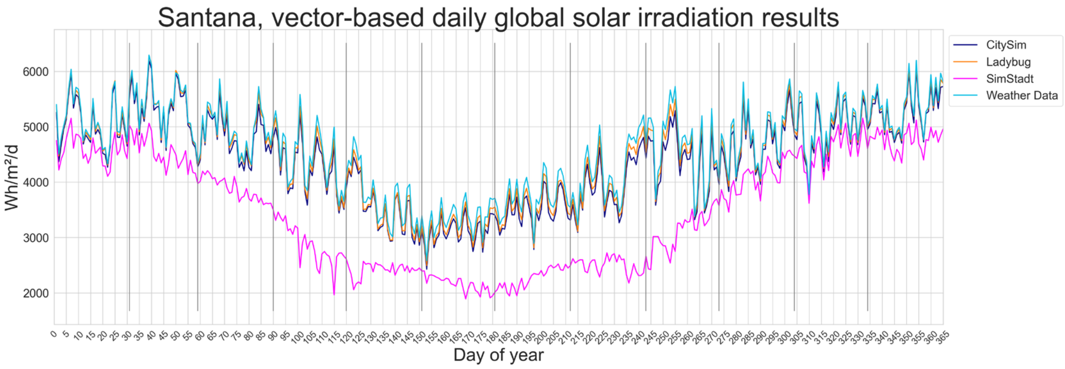


FIGURE 15 | Time series of the vector-based simulation results and the weather data for Santana.

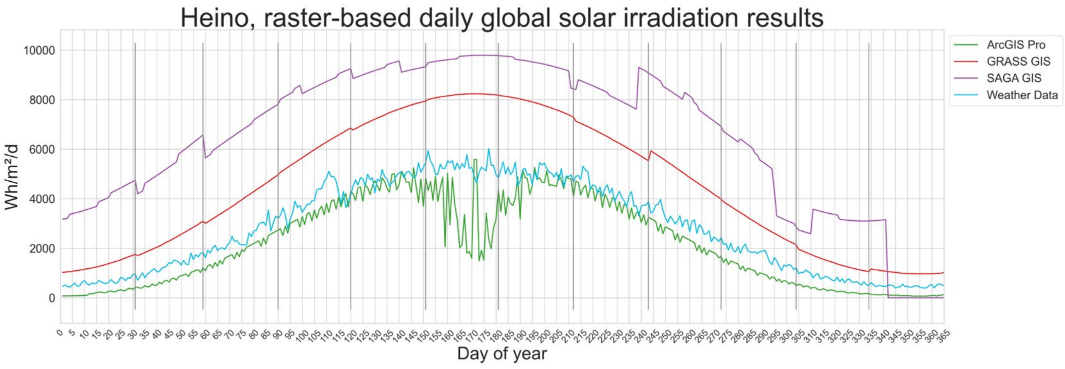


FIGURE 16 | Time series of the raster-based simulation results and the weather data for Heino.

results from both locations show the same particular behavior during the summer. In both cases, there is a significant drop in the simulated values during June (northern latitude) and December (southern latitude). This phenomenon is very well visible in both graphs (green line). The reason for this is still unclear and requires further investigation.

For the vector-based results, in the case of CitySim and Ladybug (Figure 17), their results have closer values to ground truth compared to the results in Santana. For CitySim, the yearly average difference is ~−19%; in the case of Ladybug, it yields ~−18%. As previously seen in Santana, SimStadt produces the lowest underestimation of the solar irradiation values for the vector-based simulation tools, in the case of Heino, a yearly average of − 47%.

It is relevant to remember that direct comparison of results coming from between raster-based and vector-based models can be misleading since the former adopts a clear-sky model. In contrast, the latter does not, although, in both cases, the results are expressed in Wh/m^2/d. However, from the user's point of view, we believe it is still interesting to superimpose the results in a unique graph to convey the magnitude of the differences that are to be expected when using one software tool or another, as exemplified in Figures 18 and 19 for both case studies.

4.1.1.2 | Monthly values. In a similar way to the daily values described previously, results for each simulation tool and each case study area have been aggregated and compared at the monthly level.

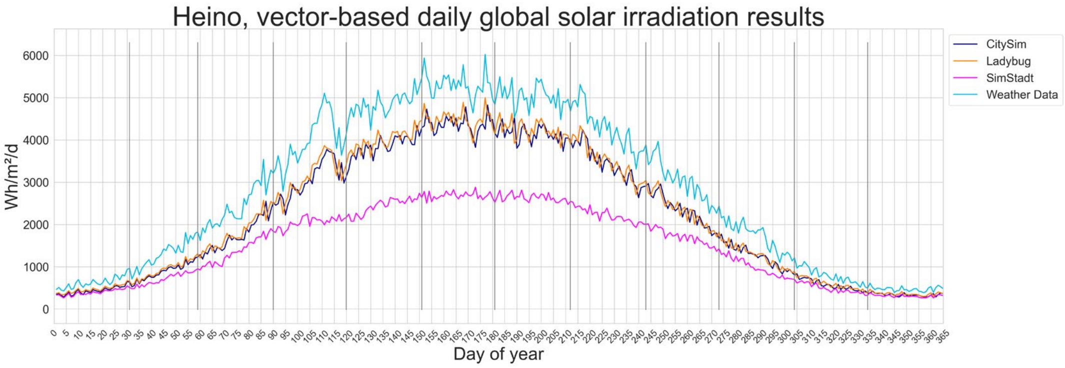


FIGURE 17 | Time series of the vector-based simulation results and the weather data for Heino.

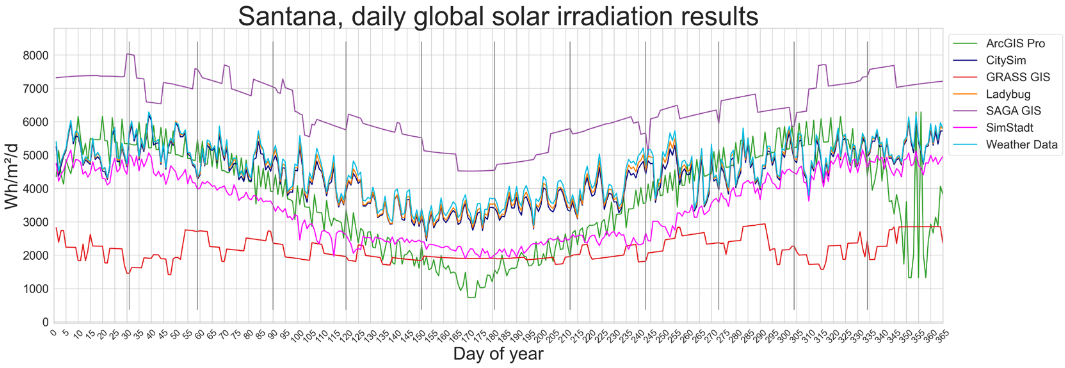


FIGURE 18 | Time series of the simulation results and the weather data for Santana.

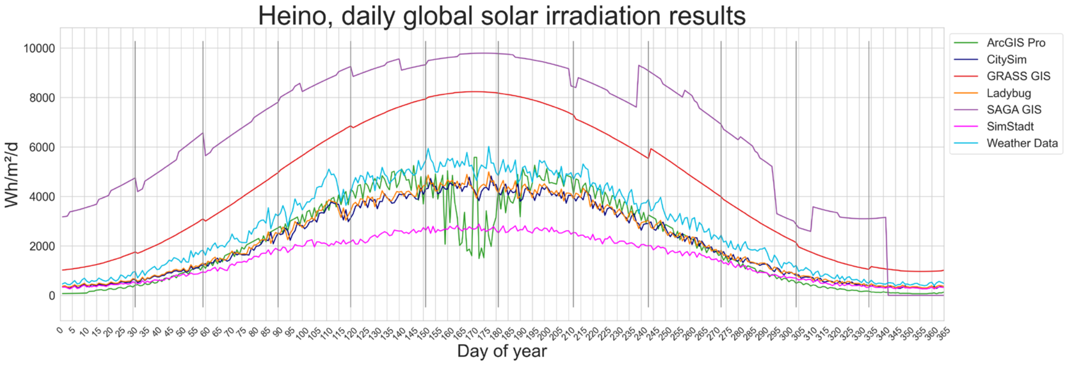


FIGURE 19 | Time series of the simulation results and the weather data for Heino.

The results for Santana are presented in Table 4. When it comes to the raster-based results, GRASS GIS produces the less accurate results on a monthly basis, with discrepancies between – 42.62% in June (winter in the southern hemisphere) up to a – 62.51% in February (summer), respectively. SAGA GIS values have an average 39.80% overestimation, with a peak of 57.50% in May. In the case of ArcGIS Pro, the results vary from a 1.65% overestimation in January up to an average 59.80% underestimation in June. When considering the RMSE values, ArcGIS Pro, SimStadt, and GRASS GIS consistently exhibit high RMSE values across all months. Notably, GRASS GIS obtains the highest RMSE value of all the software packages, reaching up to 6.52 in January.

When it comes to vector-based results, Ladybug consistently underestimates the ground truth values, exhibiting a maximum deviation of – 4.54% in June (winter) and – 0.37% in February (summer). CitySim demonstrates a similar underestimation pattern but with slightly poorer performance, displaying its most significant deviation of – 7.07% in June. SimStadt shows the highest discrepancy to ground truth with an average – 25.37% underestimation. According to the RMSE values, Ladybug achieves the lowest absolute value in February (0.03), while the highest value (0.016) correspond to July and August. CitySim demonstrates a similar performance but with slightly higher values (0.01 in February and 0.09 in September).

Table 5 shows the results for Heino. In general, some simulation tools have a similar behavior to Santana's results. For the raster-based results, SAGA GIS produces the highest results discrepancy with an average 183.89% overestimation, reaching up to 519.14% monthly average difference in January. Additionally, GRASS GIS overestimates the solar irradiation with an average 71.14% difference. This difference is higher in the winter season, specifically in December and January, with 115.16% and 108.41% difference, respectively. ArcGIS Pro discrepancies are lower compared to the other two raster-based software tools. However, it also reveals differences in the monthly average values of December (80.15%) and January (68.32%). For the vector-based results, Ladybug again underestimates the ground truth values. However, the difference is higher, with a maximum difference of – 28.56% in February. In the case of CitySim, it differs from – 17.47% in June up to – 32.06% in February. There are no changes regarding RMSE: Ladybug has the lowest values, followed closely by CitySim. For both simulation tools, the monthly values have a maximum RMSE in May.

4.1.1.3 | Annual values.

Finally, the same analysis has been conducted by further aggregating the results at the yearly level. In this case, we could also add UMEP to the comparison, as it only allows us to perform solar analysis, providing results at this level of aggregation.

Table 6 and Figure 20 contain the aggregated values of the global irradiation for Santana. Ladybug produces the closest values to the ground truth, with a difference of – 2.06%, followed by CitySim and ArcGIS Pro, with – 4.05% and – 18.9%, respectively. GRASS GIS produces the furthest values, with a difference of – 71.54%. When looking at RMSE, Ladybug produces the smallest, with a value of 0.113, followed by CitySim and SimStadt, with 0.204 and 1.265, respectively. We have no RMSE for UMEP because it only produces yearly values.

Table 7 and Figure 21 contain the aggregated values of the global irradiation for Heino. UMEP produces the closest value with a difference of 1.59%, followed by Ladybug with a difference of – 21.92%, and the lowest RMSE of 0.63. The former is followed by CitySim, with – 24.57% and 0.726, respectively. On the other hand, SAGA GIS produces the furthest results, with a difference of 79.36% and a RMSE of 3.972, followed by SimStadt, with a – 61.13% and 1.614, respectively.

Although clear-sky models should consistently deliver higher values compared to ground truth, it is interesting to note that this is not the case for the results presented in our research. The output values of ArcGIS Pro output are consistently lower, while the ones from GRASS GIS have different behavior, depending on the case study. Only SAGA GIS presents values "as expected". In the case of vector-based simulation tools output, all of them have the same behavior in both case studies.

4.1.2 | Overall Considerations on Accuracy

The results so far show that, in both study areas, SAGA GIS produces the most significant deviations when compared to the ground truth. The results in Heino show even more significant deviations than those in Santana. However, Conrad (2010) warns the users that the model based on the work of (Hofierka, Suri, and Huld 2007) "needs further revision!". Our first experiments in GRASS GIS and SAGA GIS provide outputs that follow a relatively regular bell-shaped curve (Rosenthal and Seth 2000), which is coherent with the fact that no weather data is used in their models.

For vector-based simulation tools, SimStadt, CitySim, and Ladybug use the same input weather data, albeit in different file formats. While CitySim and Ladybug produced results that were close to the measured data, this is not the case for SimStadt, as it consistently underestimated irradiation values throughout the year. The primary distinction between these tools is that SimStadt exclusively considers buildings during the simulation process. Referring to Tables 4 and 5, in both our study areas, Ladybug provides the most accurate estimation of solar irradiation, closely followed by CitySim. In selecting a simulation tool between Ladybug and CitySim, the user should consider other criteria, such as data preparation or execution time.

4.2 | Qualitative Analysis of the Simulation Tools

Table 8 shows a comparison overview of all parameters we considered for our qualitative analyses, in which we evaluate not only the accuracy of the simulation's output. Except for SimStadt, a terrain model is fundamental input data for all simulation tools.

4.3 | Raster-Based Simulation Tools Analysis

Despite our efforts to create a standardized testbed with data as "similar" as possible among all simulation tools, the comparison of results among raster-based models is still complex due to possible variations in their input parameters. While the DSM serves as a common input for these models, Esri does not mention anything about it in their documentation of ArcGIS Pro.

TABLE 4 | Santana, monthly global irradiation values from the simulation results compared to the ground truth.

	Raster-based						**Vector-based**					
	ArcGIS Pro		**GRASS GIS**		**SAGA GIS**		**CitySim**		**Ladybug**		**SimStadt**	
Month	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**
January	1.65%	0.68	−57.82%	3.02	44.25%	2.32	−1.93%	0.10	−0.63%	0.04	−10.39%	0.57
February	−2.02%	0.64	−62.51%	3.46	31.29%	1.81	−1.56%	0.09	−0.37%	0.03	−16.45%	0.95
March	−7.80%	0.63	−52.55%	2.65	44.18%	2.22	−3.26%	0.18	−1.12%	0.07	−22.24%	1.17
April	−22.47%	1.14	−53.56%	2.43	39.63%	1.87	−4.81%	0.23	−2.62%	0.13	−35.72%	1.71
May	−37.79%	1.45	−48.31%	1.85	57.50%	2.15	−5.27%	0.21	−3.84%	0.15	−34.69%	1.41
June	−59.80%	2.05	−42.62%	1.45	42.97%	1.50	−7.07%	0.24	−4.54%	0.15	−35.56%	1.25
July	−48.45%	1.85	−49.57%	1.89	36.58%	1.44	−6.90%	0.26	−4.22%	0.16	−39.08%	1.51
August	−27.17%	1.26	−52.13%	2.28	38.51%	1.73	−6.47%	0.29	−3.63%	0.16	−40.90%	1.85
September	−7.06%	0.89	−47.47%	2.31	33.83%	1.71	−5.77%	0.29	−2.70%	0.14	−32.52%	1.72
October	6.25%	0.64	−49.60%	2.49	33.89%	1.75	−3.16%	0.16	−1.39%	0.08	−15.75%	0.83
November	6.55%	0.74	−59.07%	3.04	39.40%	2.09	−2.41%	0.13	−1.01%	0.06	−9.43%	0.52
December	−36.05%	2.39	−51.34%	2.81	35.56%	1.98	−1.98%	0.12	−0.77%	0.05	−11.69%	0.67

TABLE 5 | Heino, monthly global irradiation values from the simulation results compared to the ground truth (kWh / m^2 / month).

	Raster-based						Vector-based					
	ArcGIS Pro		GRASS GIS		SAGA GIS		CitySim		Ladybug		SimStadt	
Month	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**	**Diff**	**RMSE**
January	−68.32%	0.44	108.41%	0.69	519.14%	3.29	−30.74%	0.20	−25.18%	0.17	−37.75%	0.25
February	−43.49%	0.58	79.65%	1.05	314.19%	4.13	−32.06%	0.44	−28.56%	0.39	−45.50%	0.62
March	−22.10%	0.58	62.72%	1.55	178.03%	4.35	−28.14%	0.72	−25.18%	0.63	−43.71%	1.09
April	−14.67%	0.75	50.19%	2.05	115.74%	4.64	−23.40%	0.99	−21.09%	0.86	−48.54%	2.00
May	−7.46%	0.59	52.33%	2.57	88.89%	4.34	−20.55%	1.03	−17.76%	0.87	−49.78%	2.44
June	−37.05%	2.40	54.62%	2.91	83.37%	4.42	−17.47%	0.98	−15.36%	0.82	−49.30%	2.62
July	−11.13%	0.81	55.29%	2.80	87.52%	4.41	−17.69%	0.95	−15.71%	0.80	−47.59%	2.40
August	−9.86%	0.53	49.55%	2.13	97.00%	4.19	−19.88%	0.89	−18.34%	0.79	−48.19%	2.09
September	−21.88%	0.70	65.38%	1.98	166.38%	5.00	−23.03%	0.73	−22.88%	0.69	−43.47%	1.34
October	−41.59%	0.72	74.19%	1.29	201.49%	3.60	−27.33%	0.49	−26.56%	0.46	−42.88%	0.75
November	−61.39%	0.49	86.24%	0.68	305.66%	2.40	−27.32%	0.23	−24.43%	0.20	−38.20%	0.31
December	−80.15%	0.38	115.16%	0.55	49.32%	1.32	−28.86%	0.15	−24.26%	0.12	−34.83%	0.17

TABLE 6 | Santana, yearly global irradiation values from the simulation results compared to the ground truth (kWh / m^2 / a).

	Raster-based				Vector-based		
	ArcGIS Pro	**GRASS GIS**	**SAGA GIS**	**UMEP**	**CitySim**	**Ladybug**	**SimStadt**
Total	1380.54	789.46	2326.70	1233.74	1602.42	1634.64	1268.13
Diff.	−18.90%	−71.54%	32.94%	−29.97%	−4.05%	−2.06%	−27.28%
RMSE	1.341	2.528	1.903		0.204	0.113	1.265

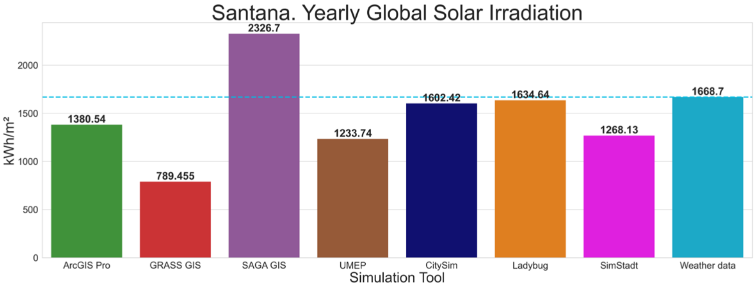


FIGURE 20 | Yearly global irradiation values obtained from the simulation results and the ground truth at Santana. The cyan dotted line depicts the reference yearly value.

TABLE 7 | Heino, yearly global irradiation values from the simulation results compared to the ground truth (kWh / m^2 / a).

	Raster-based				Vector-based		
	ArcGIS Pro	**GRASS GIS**	**SAGA GIS**	**UMEP**	**CitySim**	**Ladybug**	**SimStadt**
Total	798.16	1639.84	2384.68	1046.28	804.42	826.34	547.61
Diff.	−25.34%	45.71%	79.36%	1.59%	−24.57%	−21.92%	−61.13%
RMSE	0.904	1.875	3.972		0.726	0.63	1.614

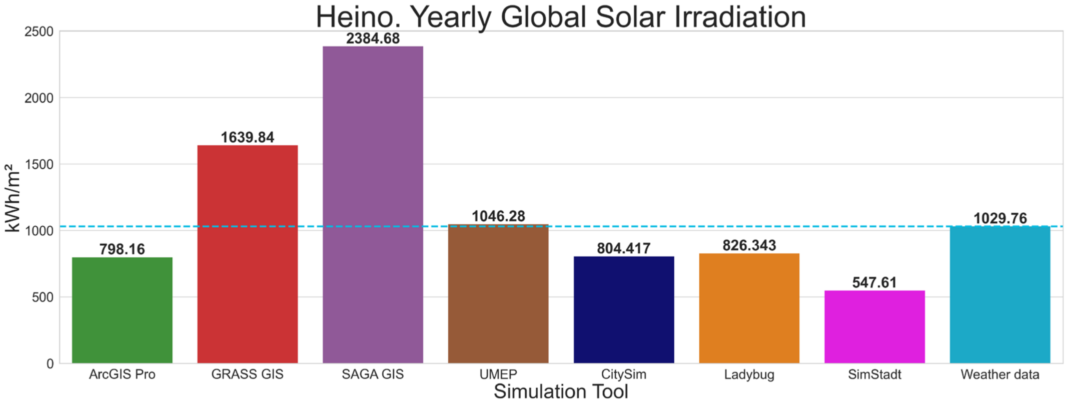


FIGURE 21 | Yearly global irradiation values obtained from the simulation results and the ground truth at Heino. The cyan dotted line depicts the reference yearly value.

The processing time in ArcGIS Pro could be improved by allowing the loading of previously computed slope and aspect raster files. However, this is not possible with the currently available software version. This restrinction force the computation these elements every time the user executes the tool, which is not efficient when testing different parameters using the same DSM. This software also lacks support for weather data.

GRASS GIS and SAGA GIS support additional parameters, like the Linke turbidity factor. UMEP has specific formatting requirements for weather data, which are fully supplied by using an Energy Plus (epw) file. The absence of a standardized ground truth for input weather data for all simulation tools affects the assessment of algorithm performance based on the location of the study area.

TABLE 8 | Comparison of the tested simulation tools for solar irradiation. Running times were calculated using a desktop computer with the following hardware specifications: Windows 11, Intel i7-9700K CPU, 64 GB RAM, AMD Radeon VII GPU and NVMe PCIe M.22280 SSD.

	Raster-based				Vector-based		
	ArcGIS Pro	**GRASS GIS**	**SAGA GIS**	**UMEP**	**CitySim**	**Ladybug**	**SimStadt**
License	Commercial	FOSS	FOSS	FOSS	FOSS	FOSS but Rhinoceros 3D/Grasshopper	FOSS
Minimum input data requirements	Raster-based DSM	Raster-based DSM	Raster-based DSM	Raster-based DSM, Weather data	Vector-based 3D scene (Buildings, relief, vegetation), weather data	Vector geometries as Brep/Mesh, weather	Vector-based 3D scene (Buildings in CityGML LoD1/2), weather
Optional input files	N/A	Slope, aspect, Linke turbidity, albedo maps	Sky View Factor, Water Vapor Pressure, Linke turbidity	Vegetation Canopy, Vegetation Trunk zone albedo value	Horizon file	N/A	N/A
Interaction	GUI/Python	GUI/Python/shell	GUI/shell	GUI/Python	GUI/shell	GUI/Python	GUI/shell
Urban features	All features represented in the DSM	All features represented in the DSM	All features represented in the DSM	Buildings, trees and ground represented in DSM	Buildings, Relief, Vegetation	Buildings, Relief, Vegetation	Buildings
Santana. Timing (HH:MM)	03:18	08:40	00:12	00:11	16:44	00:26	00:06
Heino. Timing (HH:MM)	31:34	17:19	11:51	30:24	268:45	00:18	00:11
Results: Type	2.5D surfaces	2.5D surfaces	2.5D surfaces	2.5D surfaces	3D surfaces	3D surfaces	3D surfaces
Output	Raster file	Raster file	Raster file	Raster file	TSV file	Data tree	out file
Minimum temporal resolution	Hourly	Second/daily	Hourly	Yearly	Hourly	Hourly	Hourly

One of the main limitations of the raster data format is the capability only to simulate 2.5D surfaces while excluding vertical surfaces and similar geometries, which may be indeed relevant at sites located at latitudes further away from the equator due to the sun's position over the sky. Finally, it is noteworthy to highlight the disparity in computation times: using the exact raster resolution, SAGA GIS completes the simulations faster, followed by GRASS GIS and then by ArcGIS Pro, which takes almost triple the time of SAGA GIS and double GRASS GIS.

4.4 | Vector-Based Simulation Tools Analysis

When it comes to vector-based simulation tools, Ladybug may present a steeper learning curve for GIS users due to its development within the Rhinoceros 3D/Grasshopper platform, which is more closely associated with the CAD environment rather than the GIS domain. SimStadt and CitySim share standard features such as support for CityGML, a GUI, and support for shell scripts. However, SimStadt has certain drawbacks, particularly in terms of the temporal resolution of the results, which are limited to hourly or daily values. Furthermore, to obtain the latter, users are required to activate the option *save cache files* in the GUI and subsequently navigate to the hidden folders in their operating system to locate these files. Since the temporary files do not store the GMLID of the simulated surfaces, manually identifying the corresponding RoofSurface associated with the weather station and retrieving the simulation values was done based on the simplified geometry of the building that represents the weather station. However, this manual approach is not reliable for more complex geometries.

CitySim is the slowest simulation tool. It took more than 11 days (approx. 268 h) to simulate Heino. Our input data set includes 3950 fundamental implicit geometries for trees (basically two vertical surfaces standing in an orthogonal position against each other). By removing the trees, the simulation takes ~65 h, and when using only buildings as input features, it takes 00 h:22 to complete the simulation.

5 | Conclusions

This article has presented a comparative method for conducting solar irradiation analysis utilizing several available software tools, with a focus on ensuring replicability for each tool and comparability of the results while considering data accessibility and the relief of the study area. A comparative study was conducted on seven software tools to calculate solar irradiation within an urban context. Among these tools, four adopt a raster-based approach (ArcGIS Pro, GRASS GIS, SAGA GIS, and UMEP), while the remaining three (CitySim, Ladybug, and SimStadt) use a 3D vector-based representation of the urban scene.

Our research involves data collection, including weather data obtained from meteorological stations. These data were processed to ensure the compatibility between the software tools used in this research. We evaluate these tools based on their data requirements, simulation time, and user-friendliness. Our evaluation includes factors such as the support for additional features such as buildings, vegetation, and relief in the study area. Vector-based simulations have more complex data requirements to deal with. The simulation tools require the 3D representation of urban features, including buildings, relief and vegetation, depending on the tool's requirements/possibilities. This kind of data is not easily available and causes further processing time in the data preparation steps. However, we did not include data preparation as a criteria for the selection of the software tool since this factor that varies according to the data available at the moment of performing the analyses. Therefore, the selection of the software tool can be considered based on different aspects as follows:

- Data requirements: When data access is a restriction, raster-based models are the option to go since they only require a DSM. However, not all software has identical requirements, so additional factors should be considered, such as accuracy assessment or configuration of the input parameters. Therefore, we propose the following rankings when dealing with data scarcity:
    - Simplicity of use: ArcGIS Pro, GRASS GIS/SAGA GIS, UMEP
    - Richness of configuration parameters: UMEP, GRASS GIS, SAGA GIS, ArcGIS Pro

The major disadvantage of UMEP is the periodicity of the results since the output is restricted to yearly values.

- Granularity of the results: This aspect can be related to two main groups: temporal and spatial. We mean by temporal the time resolution of the output, and despite UMEP and SimStadt, the simulation tools produce hourly values. Although there is a possibility to obtain hourly values also for from SimStadt, this is not a replicable method since the cache values have a different identification than the input data, so there is no match and further steps are thus required that may lead to errors.
    - Temporal granularity: Ladybug, CitySim, ArcGIS Pro, GRASS GIS, SAGA GIS, SimStadt, UMEP

The spatial granularity indicates the geogrpahical scale of the analysis from the results. As expected, the granularity of the raster-based methods is the spatial resolution of the input (2.5) DSM, which is not the case for vector-based methods. CitySim, Ladybug and SimStadt results reach the (3D) boundary surfaces of the corresponding feature. Therefore, these results can be used as input data for other analyses, such as the energy performance of buildings, solar panels instalation, green roofs or urban agriculture.

- Spatial granularity: Ladybug, CitySim, SimStadt, UMEP, ArcGIS Pro, GRASS GIS/SAGA GIS

- Accuracy of the results: One of the most relevant aspects for a model selection is, of course, the accuracy of the results. In our research, for the yearly values in both study areas, SAGA GIS in both cases it over estimates the ground truth values. SimStadt produces the lowest irradiance values, indicating a significant underestimation compared to the ground truth.

Looking at the timeseries plotted in Figures 18 and 19, we can say that, in very general terms, the results of all simulation tools follow the same pattern.

The results of all simulation tools follow the same pattern as the ground truth. However, there is not one simulation tool that produces the most accurate results in both locations. Ladybug, CitySim, UMEP and ArcGIS Pro show all a good accuracy in both locations against the ground truth.

Due to the yearly values presented in Tables 6 and 7, we cannot indicate which one of the analyzed simulation tools is the best for any location. Furthermore, our research lead us to suggest that in any case the calibration of the model should not be ignored (as we did on purpose for this comparative study) in order to achieve greater accuracy.

Further research could be performed to explore the impact of other parameters, such as the albedo of the surrounding elements, such as buildings, terrain, and vegetation. However, these are parameters that are not supported by all simulation tools. For example, GRASS GIS supports albedo, and SAGA GIS supports water vapor pressure, while ArcGIS Pro does not offer support for additional input data.

Additionally, further research should evaluate the quality of simulation results based on the surface inclination. We took the output values at the horizontal roof of the building located at the weather station because that surface is available for both the raster- and vector-based methods. However, wall surfaces are not represented correctly in a raster, as an intrinsic limitation of the data type. UMEP produces a file (Energyyearwall.txt) containing the wall irradiation values for "each wall column" (of the modeled raster; Lindberg et al. 2023). Their approach is based on a voxel creation method based on the input DSM resolution from the ground moving upwards. In this case, an automation tool needs to be developed to extract the values for each of the wall surfaces that belong to each building. Further solutions need to be developed to also support the inclination surfaces of buildings for the other raster-based simulation tools.

Finally, future work should also consider accuracy on vertical or tilted surfaces, as well as the possible automation to retrieve and integrate simulation results. Global irradiation values are essential for solar analyses in buildings. For example, these values are needed to calculate the solar gains of building envelopes. However, future research needs to focus on the calculation of direct and diffuse irradiation for vector-based methods. These values are fundamental for a precise analysis of PV systems. The DISC-model (Maxwell 1987) can be used to calculate direct and diffuse irradiation values based on the global horizontal data. Another possibility is the use of pvlib (Holmgren, Hansen, and Mikofski 2018), an open-source library for the simulation of the performance of PV systems. Its functionalities include the calculation of direct and diffuse irradiation for a given location.

Acknowledgments

The authors would like to thank Jérôme Kämpf (Idiap), Giuseppe Peronato, and Eric Duminil (HFT Stuttgart) for the fruitful discussions and their hints on Citysim and SimStadt, respectively.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Input data and scripts created during this research are available in a GitHub repository AXGIST4SolarSim. This repository contains the required input data as well as the Python scripts to replicate the results obtained in this research. At its root, the link contains a README file that explains each subfolder, the Python environment requirements, and all the Python scripts to execute. However, there are several considerations to mention for data sharing: (1)GitHub does not allow sharing files bigger than 100 MB, (2) GitHub does not synchronize more than 1.000 files per folder, (3) Figshare supports up to 20GB of data upload per user. Based on this, we decided to place a README file in each folder we could not place data directly in the repository. Those files contain a link to download the data from the figshare repository. About the figshare shared data:

- Folder *Input_Data/Heino/csv*: Data are compressed into 7z format to facilitate sharing, files to download are compressed up to a 86 % with sizes varying from 183.7mb to 7.71GB.
- Consolidated output files from ArcGIS Pro Pro v3.0, GRASS GIS v7.8.7 and SAGA GIS v8.5.1 of *Heino* are not included since each file has a size of ∼ 32GB per file, requiring ∼ 450GB of storage.

In total, the figshare repository storages 14.96GB. All Python scripts point to the relative path, considering the repository structure. All data from Santana are included in the GitHub repository since it is a smaller study area. Consolidated results for both study areas are included, as well as their graphs. For further documentation, please refer to the repository.

References

Agugiaro, G., F. Nex, F. Remondino, R. de, S. Droghetti, and C. Furlanello. 2012. "Solar Radiation Estimation on Building Roofs and Web-Based Solar Cadastre." *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 1: 177–182. https://doi.org/10.5194/isprsannals-I-2-177-2012.

Bensehla, S., Y. Lazri, and M. C. Brito. 2021. "Solar Potential of Urban Forms of a Cold Semi-Arid City in Algeria in the Present and Future Climate." *Energy for Sustainable Development* 62: 151–162. https://doi.org/10.1016/j.esd.2021.04.004.

Buzra, U., and E. Serdari. 2023. "A Comparison Analysis of Different PV Simulation Tools Using Satellite Data." *Electrical Engineering* 105: 1–8. https://doi.org/10.1007/s00202-023-01814-6.

Calcabrini, A., H. Ziar, O. Isabella, and M. Zeman. 2019. "A Simplified Skyline-Based Method for Estimating the Annual Solar Energy Potential in Urban Environments." *Nature Energy* 4, no. 3: 206–215. https://doi.org/10.1038/s41560-018-0318-6.

Camargo Luis, R., R. Zink, W. Dorner, and G. Stoeglehner. 2015. "Spatio-Temporal Modeling of Roof-Top Photovoltaic Panels for Improved Technical Potential Assessment and Electricity Peak Load Offsetting at the Municipal Scale." *Computers, Environment and Urban Systems* 52: 58–69. https://doi.org/10.1016/j.compenvurbsys.2015.03.002.

Chen, G., L. Rong, and G. Zhang. 2020. "Comparison of Urban Airflow Between Solar-Induced Thermal Wall and Uniform Wall Temperature Boundary Conditions by Coupling CitySim and CFD." *Building and Environment* 172: 106732. https://doi.org/10.1016/j.buildenv.2020.106732.

Coccolo, S., and J. Kämpf. 2015. "Urban Energy Simulation Based on A New Data Model Paradigm: The CityGML Application Domain Extension Energy. A Case Study in the EPFL Campus of Lausanne." Conference Paper. https://doi.org/10.26868/25222708.2015.3013.

Conrad, C. 2010. "Tool Potential Incoming Solar Radiation." SAGA-GIS Tool Library Documentation (v8.4.2). https://saga-gis.sourceforge.io/saga_tool_doc/8.4.2/ta_lighting_2.html.

de Sá, B. A., T. Dezuo, and D. Ohf. 2022. "Shadow Modelling Algorithm for Photovoltaic Systems: Extended Analysis and Simulation." *Journal of Control, Automation and Electrical Systems* 33, no. 5: 1507–1518. https://doi.org/10.1007/s40313-022-00905-2.

Duminil, E., K. Brassel, R. Nouvel, et al. 2022. "SimStadt." Accessed March 1, 2024. https://simstadt.hft-stuttgart.de/.

Esri. 2024. "Area Solar Radiation (Spatial Analyst)." Accessed March 1, 2024. https://pro.arcgis.com/en/proapp/latest/tool-reference/spatial-analyst/area-solar-radiation.htm.

EU Science Hub. 2022. "PVGIS Online Tool." Accessed March 1, 2024. https://joint-research-centre.ec.europa.eu/pvgis-online-tool_en.

Freitas, J. D. S., J. Cronemberger, R. M. Soares, and C. N. D. Amorim. 2020. "Modeling and Assessing BIPV Envelopes Using Parametric Rhinoceros Plugins Grasshopper and Ladybug." *Renewable Energy* 160: 1468–1479. https://doi.org/10.1016/j.renene.2020.05.137.

Freitas, S., C. Catita, P. Redweik, and M. C. Brito. 2015. "Modelling Solar Potential in the Urban Environment: State-Ofthe-Art Review." *Renewable and Sustainable Energy Reviews* 41: 915–931. https://doi.org/10.1016/j.rser.2014.08.060.

Garegnani, G., P. Zambelli, G. Grilli, and D. Vettorato. 2015. "Evaluation of Wind, Solar and Hydro Energy Potential Using GRASS." Conference Paper.

GeoSampa. 2023. "Mapa Digital da Cidade de S~ao Paulo." Accessed March 1, 2024. https://geosampa.prefeitura.sp.gov.br/PaginasPublicas/_SBC.aspx.

Giannelli, D. 2021. "Solar Analysis on Buildings of Favelas in Sao Paulo to Estimate PV Potential." English. MA Thesis, Delft, NL: Delft University of Technology. Accessed March 1, 2024. http://resolver.tudelft.nl/uuid:05ff635a-8113-4d65-aae1-bc0c727831d8.

Gonçalves, J. E., H. Montazeri, T. van, and D. Saelens. 2021. "Performance of Building Integrated Photovoltaic Facades: Impact of Exterior Convective Heat Transfer." *Applied Energy* 287: 116538. https://doi.org/10.1016/j.apenergy.2021.116538.

Groenewolt, A., J. Bakker, J. Hofer, Z. Nagy, and A. Schlueter. 2016. "Methods for Modelling and Analysis of Bendable Photovoltaic Modules on Irregularly Curved Surfaces." *International Journal of Energy and Environmental Engineering* 7, no. 3: 261–271. https://doi.org/10.1007/s40095-016-0215-3.

Gröger, G., T. Kolbe, C. Nagel, and K. Häfele. 2012. "OGC City Geography Markup Language (CityGML) Encoding Standard." OGC 12-019. Open Geospatial Consortium. https://www.opengeospatial.org/standards/citygml.

Gulben, E. J. G., J. M. Ling, D. Gulben, and N. R. Estoperez. 2019. "Solar Power Resource Assessment Using Light Detection and Ranging Data and Open Source Geographic Information System." In *2019 IEEE Milan PowerTech*, 1–6. Milan, Italy: IEEE. https://doi.org/10.1109/PTC.2019.8810610. Accessed March 1, 2024. https://ieeexplore.ieee.org/document/8810610/.

Hassaan, M. A., A. Hassan, and H. Al-Dashti. 2021. "GIS-Based Suitability Analysis for Siting Solar Power Plants in Kuwait." *Egyptian Journal of Remote Sensing and Space Science* 24, no. 3: 453–461. https://doi.org/10.1016/j.ejrs.2020.11.004.

Hofierka, J., M. Suri, and T. Huld. 2007. "r.sun." GRASS GIS 8.2.2dev Reference Manual. https://grass.osgeo.org/grass82/manuals/r.sun.html.

Holmgren, W. F., C. W. Hansen, and M. A. Mikofski. 2018. "Pvlib Python: A Python Package for Modeling Solar Energy Systems." *Journal of Open Source Software* 3, no. 29: 884. https://doi.org/10.21105/joss.00884.

Izquierdo, S., M. Rodrigues, and N. Fueyo. 2008. "A Method for Estimating the Geographical Distribution of the Available Roof Surface Area for Large-Scale Photovoltaic Energy-Potential Evaluations." *Solar Energy* 82, no. 10: 929–939. https://doi.org/10.1016/j.solener.2008.03.007.

Jochem, A., V. Wichmann, and B. Höfle. 2009. "Large Area Point Cloud Based Solar Radiation Modeling." *Hamburger Beiträge Zur Physischen Geographie Und Landschaftsökologie* 21: 9.

Kaemco. 2023. "kaemco—Download." Accessed March 1, 2024. http://kaemco.ch/download.php.

Kausika, B., and W. Van Sark. 2021. "Calibration and Validation of ArcGIS Solar Radiation Tool for Photovoltaic Potential Determination in The Netherlands." *Energies* 14, no. 7: 1865. https://doi.org/10.3390/en14071865.

KNMI. 2024. "Automatic Weather Stations." Accessed March 1, 2024. https://www.knmi.nl/kennis-endatacentrum/uitleg/automatische-weerstations.

Kumar, N. M., S. Chakraborty, S. K. Yadav, J. Singh, and S. S. Chopra. 2022. "Advancing Simulation Tools Specific to Floating Solar Photovoltaic Systems – Comparative Analysis of Field-Measured and Simulated Energy Performance." *Sustainable Energy Technologies and Assessments* 52: 102168. https://doi.org/10.1016/j.seta.2022.102168.

Ladybug Tools LLC. 2022. "Ladybug Tools." Accessed March 1, 2024. https://www.ladybug.tools/ladybug.html.

Lawrie, L. K., and D. B. Crawley. 2023. "Development of Global Typical Meteorological Years (TMYx).". http://climate.onebuilding.org.

Liang, J., J. Gong, X. Xie, and J. Sun. 2020. "Solar3D: An Open-Source Tool for Estimating Solar Radiation in Urban Environments." *ISPRS International Journal of Geo-Information* 9: 9. https://doi.org/10.3390/ijgi9090524.

Lindberg, F., T. Sun, S. Grimmond, Y. Tang, and N. Wallenberg. 2023. "UMEP Manual—UMEP Manual Documentation." Accessed March 1, 2024. https://umep-docs.readthedocs.io/en/latest/index.html.

Maxwell, E. L. 1987. "A Quasi-Physical Model for Converting Hourly Global Horizontal to Direct Normal Insolation." Solar Energy Research Institute, SERI/TR-215-3087, pp. 35–46. http://rredc.nrel.gov/solar/pubs/PDFs/TR-215-3087.pdf.

Milosavljević, D. D., T. S. Kevkić, and S. J. Jovanović. 2022. "Review and Validation of Photovoltaic Solar Simulation Tools/Software Based on Case Study." *Open Physics* 20, no. 1: 431–451. https://doi.org/10.1515/phys-2022-0042.

Mohajeri, N., G. Upadhyay, A. Gudmundsson, D. Assouline, J. Kämpf, and J.-L Scartezzini. 2016. "Effects of Urban Compactness on Solar Energy Potential." *Renewable Energy* 93: 469–482. https://doi.org/10.1016/j.renene.2016.02.053.

Mujić, N., and A. Karabegović. 2023. "Calculating and Comparing Solar Radiation Results Using GIS in the City Sarajevo Area." Conference Paper, pp. 285–290. Accessed March 1, 2024. annals-csis.org/proceedings/2023/drp/6245.html.

Mutani, G., S. Coccolo, J. Kaempf, and M. Bilardo. 2018. *CitySim Guide: Urban Energy Modelling*. Torino: CreateSpace Independent Publishing Platform. https://www.amazon.com/CitySim-Guide-Urban-Energy-Modelling/dp/1987609735.

Neteler, M., and H. Mitasova. 2010. *Open Source GIS: A GRASS GIS Approach*. 3rd ed. New York, NY: Springer. The International Series in Engineering and Computer Science 773.

Perez, R., R. Seals, P. Ineichen, R. Stewart, and D. Menicucci. 1987. "A New Simplified Version of the Perez Diffuse Irradiance Model for Tilted Surfaces." *Solar Energy* 39, no. 3: 221–231. https://doi.org/10.1016/S0038-092X(87)80031-2.

Peters, R., B. Dukai, S. Vitalis, J. van Liempt, and J. Stoter. 2022. “Automated 3D Reconstruction of LoD2 and LoD1 Models for all 10 Million Buildings of The Netherlands.” *Photogrammetric Engineering and Remote Sensing* 88, no. 3: 165–170. https://doi.org/10.14358/PERS.21-00032R2.

Polo, J., and R. J. García. 2023. “Solar Potential Uncertainty in Building Rooftops as a Function of Digital Surface Model Accuracy.” *Remote Sensing* 15, no. 3: 567. https://doi.org/10.3390/rs15030567.

Prieto, I., J. L. Izkara, and E. Usobiaga. 2019. “The Application of LiDAR Data for the Solar Potential Analysis Based on Urban 3D Model.” *Remote Sensing* 11, no. 20: 2348.

Ritchie, H., M. Roser, and P. Rosado. 2022. *Energy*. Accessed March 1, 2024. https://ourworldindata.org/energy#article-citation.

Rodríguez, L. R., E. Duminil, J. S. Ramos, and U. Eicker. 2017. “Assessment of the Photovoltaic Potential at Urban Level Based on 3D City Models: A Case Study and New Methodological Approach.” *Solar Energy* 146: 264–275. https://doi.org/10.1016/j.solener.2017.02.043.

Rodríguez, L. R., R. Nouvel, E. Duminil, and U. Eicker. 2017. “Setting Intelligent City Tiling Strategies for Urban Shading Simulations.” *Solar Energy* 157: 880–894. https://doi.org/10.1016/j.solener.2017.09.017.

Rosenthal, J. S., and J. Seth. 2000. *A First Look at Rigorous Probability Theory*. River Edge, NJ: World Scientific. https://search.library.wisc.edu/catalog/9910238651302121.

Rostami, E., N. Nasrollahi, and J. Khodakarami. 2024. “A Comprehensive Study of How Urban Morphological Parameters Impact the Solar Potential, Energy Consumption and Daylight Autonomy in Canyons and Buildings.” *Energy and Buildings* 305: 113904. https://doi.org/10.1016/j.enbuild.2024.113904.

Roudsari, M. S., and M. Pak. 2013. “Ladybug: A Parametric Environmental Plugin for Grasshopper to Help Designers Create an Environmentally-Conscious Design.” In *Building Simulation Conference Proceedings*, 3128–3135. Chambery France: IBPSA Publications. https://doi.org/10.26868/25222708.2013.2499.

Safe Software. 2023. “About Workspaces.” Accessed March 1, 2024. https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Desktop/Workbench/workspaces_about.htm.

Schumacher, J. 2014. “What is INSEL?,” Accessed May 22, 2023. https://insel.eu/en/what-isinsel.html.

SoDa. 2010. “Linke Turbidity (TL) Factor Worldwide.” Accessed March 1, 2024. https://www.sodapro.com/help/general-knowledge/linke-turbidity-factor.

Stuurgroep AHN. 2019. “Het Actueel Hoogtebestand Nederland (AHN3).” Accessed March 1, 2024. https://app.pdok.nl/ahn3-downloadpage/.

UN Population Division. 2018. *The World's Cities in 2018*. Technical Report, 34. New York, NY: United Nations. https://digitallibrary.un.org/record/3799524.

Viana-Fons, J. D., J. Gonzálvez-Maciá, and J. Payá. 2020. “Development and Validation in a 2D-GIS Environment of a 3D Shadow Cast Vector-Based Model on Arbitrarily Orientated and Tilted Surfaces.” *Energy and Buildings* 224: 110258. https://doi.org/10.1016/j.enbuild.2020.110258.

Weyrer, T. N. 2012. “Solar Potential Analysis of Roof Surfaces in Baton Rouge, Louisiana, Using GRASS GIS.”

Wilcox, S., and W. Marion. 2008. “Users Manual for TMY3 Data Sets. Technical Report.” NREL/TP-581-43156. Colorado, USA: National Reneweable Energy Laboratory. Accessed March 1, 2024. https://www.nrel.gov/docs/fy08osti/43156.pdf.

Wong, M. S., R. Zhu, Z. Liu, et al. 2016. “Estimation of Hong Kong's Solar Energy Potential Using GIS and Remote Sensing Technologies.” *Renewable Energy* 99: 325–335. https://doi.org/10.1016/j.renene.2016.07.003.

World Weather Online. 2024. “World Weather API and Weather Forecast.” Accessed March 1, 2024. https://www.worldweatheronline.com/.

Würstle, P., T. Santhanavanich, R. Padsala, and V. Coors. 2020. “The Conception of an Urban Energy Dashboard Using 3D City Models.” In: Proceedings of the Eleventh ACM International Conference on Future Energy Systems, Virtual Event Australia, ACM, pp. 523–527. https://doi.org/10.1145/3396851.3402650.

Yuzhen, J., and C. León-Sánchez. 2023. “Dynamic Energy Simulations Based on the 3D BAG v2.0. Original-Date: 2023-01-17T11:26:29Z.” Accessed March 1, 2024. https://github.com/tudelft3d/Dynamic-energy-simulations-based-on-the-3D-BAG-2.0.

Zhu, R., M. S. Wong, L. You, and P. Santi. 2020. “The Effect of Urban Morphology on the Solar Capacity of Three-Dimensional Cities.” *Renewable Energy* 153: 1111–1126. https://doi.org/10.1016/j.renene.2020.02.050.